INFRASTRUCTURE FOR ELECTRONIC COMMERCE

A

Regardless of their basic purpose, virtually all e-commerce sites rest on the same network structures, communication protocols, and Web standards—an infrastructure that originated over 30 years ago. In this appendix, we briefly review the structures, protocols, and standards underlying the millions of sites used to sell to, service, and communicate with both customers and business partners. We also examine the infrastructure of some newer network applications, including streaming media and peer-to-peer (P2P) networks.

A NETWORK OF NETWORKS

Although many use the Web and the Internet on a daily basis, few have a clear understanding of its basic operation. From a physical standpoint, the Internet is a network of thousands of interconnected networks. Included among the interconnected networks are (1) the interconnected *backbones*, which have international reach; (2) a multitude of access/delivery subnetworks; and (3) thousands of private and institutional networks connecting various organizational servers that contain much of the information of interest.

The backbones are run by the **network service providers (NSPs)**, which are the major telecommunication companies, such as MCI and Sprint, that maintain and service the Internet's high-speed backbones. Each backbone handles hundreds of terabytes of information per month. The delivery subnetworks are provided by the local and regional **Internet service providers (ISPs)**. ISPs exchange data with the NSPs at **network access points (NAPs)**. Pacific Bell NAP (San Francisco) and Ameritech NAP (Chicago) are examples of such exchange points.

network service providers (NSPs)
Major telecommunication companies, such as MCI and Sprint, that maintain and service the Internet's high-speed backbones.

Internet service providers (ISPs)
Companies that provide Internet delivery subnetworks at the local and regional level.

network access point (NAP)
An intermediate network exchange point that connects ISPs to NSPs.

When a user issues a request on the Internet from a computer, the request will likely traverse an ISP network, move over one or more of the backbones, and then travel across another ISP network to the computer containing the information of interest. The response to the request will follow a similar sort of path. For any given request and associated response, there is no preset route. In fact, the request and response are each broken into smaller segments called **packets**, and the packets can follow different paths. Special computers called **routers** determine the paths traversed by the packets. Routers have updateable maps of the networks on the Internet that enable them to determine the paths for the packets. Cisco (cisco.com) is one of the premier providers of high-speed routers.

packets
Small segments of messages sent over the Internet; each packet contains both data from and the addresses of the sending and receiving computers.

routers
Special computers that determine the paths traversed by data packets across the Internet.

One factor that distinguishes the various networks and subnetworks is their speed, or **bandwidth**. The bandwidths of digital networks and communication devices are rated in bits per second (bps). Today, most consumers connect to the Internet over the telephone through digital modems at 56 Kbps (kilobits per second). In some residential areas or at work, users have access to higher-speed connections. The number of homes, for example, with *digital subscriber line (DSL)* connections or cable connections is rapidly increasing. DSL connections run at 1 to 1.5 Mbps (megabits per second), whereas cable connections offer speeds of up to 10 Mbps. A megabit equals 1 million bits. Many businesses are connected to their ISPs via a T-1 digital circuit. Students at many universities enjoy this sort of connection (or something faster). The speed of a T-1 line is 1.544 Mbps. The speeds of various Internet connections are summarized in Exhibit A.1 (next page).

bandwidth
The speed at which content can be delivered across a network; it is rated in bits per second (bps).

You have probably heard the old adage that a chain is only as strong as its weakest link. On the Internet, the weakest link is the "last mile," or the connection between a residence or business and an ISP. At 56 Kbps, downloading anything but a standard Web page is a tortuous exercise. A standard Web page with text and graphics is around 400 kilobits. With a 56K modem, it takes about 7 seconds to retrieve the page. A cable modem takes about 0.04 seconds. The percentage of residences in the world with broadband connections (e.g., cable or DSL) is still very low, although that is changing rapidly. In the United States, about 13 percent of residences have broadband connections (Beardsley 2003). In South Korea, the leader in residential broadband, 50 percent of residences have broadband connections. Obviously, the lack of broadband residential connections is a major impediment for e-commerce sites that use advanced multimedia or streaming audio and video technologies, which require cable modem or T-1 speeds.

INTERNET PROTOCOLS

One thing that amazes people about the Internet is that no one is officially in charge. It is not like the international telephone system, which is operated by a small set of very large companies and regulated by national governments. This is one of the reasons that enterprises were initially reluctant to use the Internet for business purposes.

EXHIBIT A.1 **Bandwidth Specifications**

Technology	Speed	Description	Application
Digital modem	56 Kbps	Data over public telephone networks	Dial-up connections
ADSL (asynchronous digital subscriber line)	1.5 to 8.2 Mbps	Data over public telephone networks	Residential and commercial hookups
Cable modem	1 to 10 Mbps	Data over the cable network	Residential hookups
T-1	1.544 Mbps	Dedicated digital circuit	Company backbone to ISP
T-3	44.736 Mbps	Dedicated digital circuit	ISP to Internet infrastructure; smaller links in Internet infrastructure
OC-3	155.52 Mbps	Optical fiber carrier	Large company backbone to Internet backbone
OC-12	622.08 Mbps	Optical fiber carrier	Internet backbone
OC-48	2.488 Gbps	Optical fiber carrier	Internet backbone; this is the speed of the leading edge networks (e.g., Internet2)
OC-96	4.976 Gbps	Optical fiber carrier	Internet backbone

Internet Council for Assigned Names and Numbers (ICANN)
Nonprofit organization that manages various technical and policy issues relating to the Internet that require central coordination; it has no regulatory or statutory power.

The closest thing the Internet has to a ruling body is the **Internet Council for Assigned Names and Numbers (ICANN)**. ICANN (icann.org) is a nonprofit organization that was formed in 1998. Prior to ICANN, the coordination of the Internet was handled on an ad hoc and volunteer basis. This informality was the result of the culture of the research community that originally developed the Internet. The growing business and international use of the Internet necessitated a more formal and accountable structure that reflected the diversity of the user community. ICANN has no regulatory or statutory power. Instead, it oversees the management of various technical and policy issues that require central coordination. Cooperation with those policies is voluntary. Over time, ICANN has resumed responsibility for four key areas: the Domain Name System (DNS); the allocation of IP address space; the management of the root server system; and the coordination of protocol number assignment. All four of these areas form the base around which the Internet is built.

A recent survey, published in January 2003 by the Internet Software Consortium (isc.org), revealed that there were over 171 million host computers on the Internet in 230 countries (ISC 2003). Clearly, not all of these computers are the same. The problem is: How are these different computers interconnected in such a way that they form the Internet? Loshin (1997) states the problem this way:

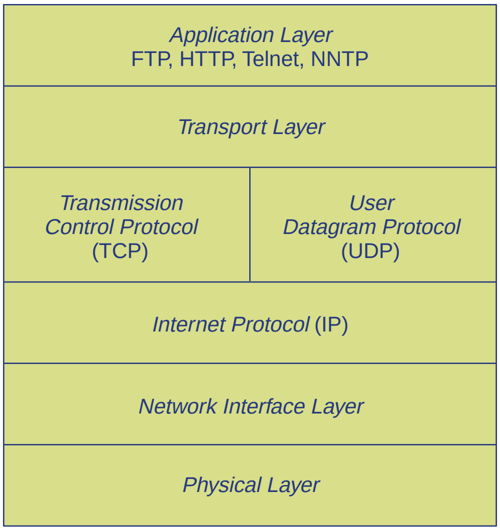
> *The problem of internetworking is how to build a set of protocols that can handle communications between any two (or more) computers, using any type of operating system, and connected using any kind of physical medium. To complicate matters, we can assume that no connected system has any knowledge about the other systems: There is no way of knowing where the remote system is, what kind of software it uses, or what kind of hardware platform it runs on.*

protocol
A set of rules that determine how two computers communicate with one another over a network.

A **protocol** is a set of rules that determine how two computers communicate with one another over a network. The protocols around which the Internet was and still is designed embody a series of design principles (Treese and Stewart 1998):

- **Interoperability.** The system supports computers and software from different vendors. For e-commerce, this means that customers or businesses do not require specific systems to conduct business.
- **Layered.** The Internet protocols work in layers, with each layer building on the layers at lower levels. This layered architecture is shown in Exhibit A.2.
- **Simple.** Each of the layers in the architecture provides only a few functions or operations. This means that application programmers are hidden from the complexities of the underlying hardware.
- **End-to-end.** Interpretation of the data happens at the application layer (i.e., the sending and receiving side) and not at the network layers. It is much like the post office. The job of the post office is to deliver mail; only the sender and receiver are concerned about the contents of the envelope.

EXHIBIT A.2 TCP/IP Architecture



TCP/IP

The protocol that solves the global internetworking problem is the **Transmission Control Protocol/Internet Protocol (TCP/IP)**. Any computer or system connected to the Internet runs TCP/IP. This is the only thing that all of the computers and systems on the Internet share in common.

Actually, as shown in Exhibit A.2, TCP/IP is two protocols—TCP and IP. TCP ensures that two computers can communicate with one another in a reliable fashion. Each TCP communication must be acknowledged as received. If the communication is not acknowledged in a reasonable time, then the sending computer must retransmit the data. In order for one computer to send a request or a response to another computer on the Internet, the request or response must be divided into packets that are labeled with the addresses of the sending and receiving computers. This is where IP comes into play. IP formats the packets and assigns addresses.

The current version of IP is **IP version 4 (IPv4)**. Under this version, Internet addresses are 32 bits long and written as four sets of numbers separated by periods (e.g., 130.211.100.5). This format is also called **dotted quad addressing**. You are probably familiar with addresses in the format of www.yahoo.com. Behind every one of these English-like addresses is a 32-bit numerical address.

With IPv4, the maximum number of available addresses is slightly over 4 billion (2^{32}). This sounds like a large number. However, with the coming generation of mobile and wireless devices, the growing number of Internet users in developing countries, and the increasing use of peer-to-peer applications, IP addresses will eventually be in short supply.

For this and other reasons, various Internet policy and standards boards are working on the **next generation Internet Protocol (IPng)**. This protocol goes by the name of **IP version 6 (IPv6)**. IPv6 is designed to improve upon IPv4's scalability, security, ease of configuration, and network management. IPv6 uses 128-bit addresses. This will allow 1 quadrillion computers (10^{15}) to be connected to the Internet. Under this scheme, one can imagine individual homes having their own networks. These home networks could be used to interconnect and access not only PCs within the home, but also a wide range of appliances, each with their own unique address.

DOMAIN NAMES

Names such as "www.microsoft.com" that reference particular computers on the Internet are called **domain names**. Domain names are divided into segments separated by periods. The part on the very left is the name of the specific computer, the part on the very right is the top-level domain to which the computer belongs, and the parts in between are the subdomains. In the case of "www.microsoft.com" the specific computer is "www," the top-level domain is "com," and the subdomain is "microsoft." Domain names are organized in a hierarchical fashion. At the top of the hierarchy is a root domain. Below the root are the top-level domains, which originally included .com, .edu, .gov, .mil, .net, .org, and .int. Of these, the .com, .net, and .edu domains represent the vast majority (109 million out of 171 million) of the names (ISC 2003). Below each top-level domain is the next layer of subdomains, below that is another layer of subdomains, and so on. The leaf nodes of the hierarchy are the actual computers.

Transmission Control Protocol/Internet Protocol (TCP/IP)
Two combined protocols that together solve the problem of global internetworking by ensuring that two computers can communicate with each other reliably; each TCP communication must be acknowledged as received or the sending computer will retransmit the message.

IP version 4 (IPv4)
The current version of Internet Protocol, under which Internet addresses are 32 bits long and written as four sets of numbers separated by periods.

dotted quad addressing
The format in which Internet addresses are written as four sets of numbers separated by periods.

next generation Internet Protocol (IPng)
Protocol that, when implemented, will improve upon IPv4's scalability, security, ease of configuration, and network management.

IP version 6 (IPv6)
Version of the Internet Protocol, still in the planning stage, that will replace IPv4 and improve network management.

domain name
The name used to reference particular computers on the Internet; the name is divided into segments separated by periods.

When a user wishes to access a particular computer, they usually do so either explicitly or implicitly through the domain name, not the numerical address. Behind the scenes, a special server called the domain name server converts the name to its associated numerical. Each organization provides at least two domain servers, a primary server and a secondary server to handle overflow. If the primary or secondary server cannot resolve the name, the name is passed to the root server and then on to the appropriate top-level server (e.g., if the address is "www.microsoft.com," then it goes to the .com domain name server). The top-level server has a list of servers for the subdomains. It refers the name to the appropriate subdomain, and so on down the hierarchy until the name is resolved. Although several domain name servers might be involved the process, the whole process usually takes microseconds.

As noted earlier, ICANN coordinates the policies that govern the DNS. Originally, Network Solutions, Inc., was the only organization with the right to issue and administer domain names for most of the top-level domains. A great deal of controversy surrounded its government-granted monopoly of the registration system. As a result, ICANN signed a memorandum of understanding with the U.S. Department of Commerce that resolved the issue and allowed ICANN to grant registration rights to other private companies. A number of other companies are now accredited registrars (e.g., America Online, CORE, France Telecom, Melbourne IT, and Register.com).

Anyone can apply for a domain name. Obviously, the names that are assigned must be unique. The difficulty is that across the world several companies and organizations have the same name. Think how many companies in the United States have the name "ABC." In addition to the television broadcasting company, stores such as ABC Appliances also use "ABC." However, there can only be one "www.abc.com." Names are issued on a first-come, first-served basis. The applicant must affirm that they have the legal right to use the name. If disputes arise, the disputes are settled by ICANN's Uniform Domain Name Dispute Resolution Policy or settled in court.

Internet2
The next generation of the Internet; it will create a network for the national research community, enable revolutionary Internet applications, and ensure the rapid transfer of new network services and applications to the broader Internet community.

gigapops
The regional, high-capacity aggregation points for traffic on Internet2's leading-edge network.

The Quilt
Collaborative effort by leading U.S. research and education networking organizations to promote the development and delivery of advanced networking services to the broadest possible community by providing network services to the universities in Internet2 and to thousands of other educational institutions.

Next Generation Internet (NGI)
Consortium initiated and sponsored by the U.S. government to link government research agencies, universities, and national labs over high-speed networks, promote experimentation with the next generation of networking technologies, and demonstrate new public policy applications requiring high-speed networks.

NEW WORLD NETWORK: INTERNET2 AND NEXT GENERATION INTERNET

It is hard to determine, or even comprehend, the vast size of the Web. Some sources estimate that the *deep Web*—the reservoir of Internet content that is unreachable by the traditional search engines—contains 550 billion individual documents (Harris 2002). Whether this figure is exactly right is unimportant. What is important is that Web content continues to grow at a very rapid pace. Unfortunately, the current data infrastructures and protocols were not designed to handle this amount of data traffic for the current number of Web users. Two consortia, as well as various telecommunication companies and other companies, have spent the last few years constructing the next generation Internet.

The first of these consortia is the University Corporation for Advanced Internet Development (UCAID, ucaid.edu). UCAID is a nonprofit consortium of over 202 universities working in partnership with industry and government. Currently, they have three major initiatives underway: Internet2, Abilene, and The Quilt.

The primary goals of **Internet2** are to:

- Create a leading-edge network capability for the national research community
- Enable revolutionary Internet applications
- Ensure the rapid transfer of new network services and applications to the broader Internet community

Internet2's leading-edge network is based on a series of interconnected **gigapops**, the regional, high-capacity points of presence that serve as aggregation points for traffic from participating organizations. In turn, these gigapops are interconnected by a very high performance backbone network infrastructure. Included among the high-speed links are Abilene, vBNS, CA*net3, and many others. Internet2 utilizes IPv6. The ultimate goal of Internet2 is to connect universities so that a 30-volume encyclopedia can be transmitted in less than a second and to support applications such as distance learning, digital libraries, video conferencing, virtual laboratories, and the like.

The Abilene initiative is focused on creating a 10-gigabit-per-second national backbone supporting high-performance connectivity and Internet innovation within the U.S. research university community. The backbone is designed to service applications such as Internet-based High Definition Television and remote control of distant telescopes.

An initiative called The Quilt was announced by UCAID in October 2001. **The Quilt** involves over 22 leading research and education networking organizations in the United States. The goal of this initiative is to promote the development and delivery of advanced networking services to the broadest possible community. The group provides network services to the universities in Internet2 and to thousands of other educational institutions

The second consortium involved in developing the new network world was the government initiated and sponsored **Next Generation Internet (NGI)**. Started by the Clinton administration, this initiative included government research agencies such as the Defense Advanced Research Projects Agency

(DARPA), the Department of Energy, the National Science Foundation (NSF), the National Aeronautics and Space Administration (NASA), and the National Institute of Standards and Technology (NIST). The primary aims of the consortium were to:

- Connect universities and national labs with high-speed networks that were 100 to 1,000 times faster than today's Internet
- Promote experimentation with the next generation of networking technologies
- Demonstrate new applications requiring high-speed networks such as health care, national security, energy research, biomedical research, and environmental monitoring

Except for meeting the goal of terabit-per-second networking in fiscal year 2002, the NGI initiative was successfully completed at the end of 2001. Advanced network research programs are now being coordinated by the Large Scale Networking (LSN) Coordinating Group (itrd.gov).

Just as the original Internet came from efforts sponsored by NSF and DARPA, it is believed that the research being done by UCAID and LSN will ultimately benefit the public. While they will certainly impact the bandwidth among the major nodes of the Internet, it still has not eliminated the transmission barriers across the last mile to most homes and businesses.

INTERNET CLIENT/SERVER APPLICATIONS

To end users, the lower-level protocols such as TCP/IP on which the Internet rests are obscured. Users do not "see" these protocols as they go about their business on the Internet. Instead, end users interact with the Internet through one of several client/server applications. As the name suggests, a client/server application has two major classes of software:

- Client software usually resides on an end user's desktop and provides navigation and display.
- Server software usually resides on a workstation or server class machine and provides backend data access services (where the data can be something as simple as a file or as complex as a relational database).

The most widely used client/server applications on the Internet are listed in Exhibit A.3. As noted in Exhibit A.3, each of these applications rests on one or more protocols that define how the clients and servers communicate with one another.

WEB-BASED CLIENT/SERVER APPLICATIONS

The vast majority of e-commerce applications are Web based. In a Web-based application, the clients are called Web browsers, and the servers are simply called Web servers. Like other client/server applications, Web browsers and servers need (1) an addressing scheme so they can locate one another and send requests and responses back and forth and (2) a communication protocol so they can communicate with one another.

EXHIBIT A.3 Internet Client/Server Applications

Application	Protocol(s)	Purpose
E-mail	Simple Mail Transport Protocol (SMTP) Post Office Protocol version 3 (POP3) Multipurpose Internet Mail Extensions (MIME)	Allows the transmission of text messages and binary attachments across the Internet.
File transfer	File Transfer Protocol (FTP)	Enables files to be uploaded and downloaded across the Internet.
Chat	Internet Relay Chat (IRC)	Provides a way for users to talk to one another in real time over the Internet. The real-time chat groups are called *channels*.
Usenet newsgroups	Network News Transfer Protocol (NNTP)	Discussion forums where users can asynchronously post messages and read messages posted by others.
World Wide Web (Web)	Hypertext Transport Protocol (HTTP)	Offers access to hypertext documents, executable programs, and other Internet resources.

Uniform Resource Locator (URL)
The addressing scheme used to locate documents on the Web.

hypertext transport protocol (HTTP)
A lightweight communication protocol that enables Web browsers and Web servers to converse with one another; of its seven commands, GET and POST make up the majority of requests issued by browsers.

The addressing scheme used on the Web is the **Uniform Resource Locator (URL)**. We are all familiar with addresses such as "www.anywhere.com/web_page.htm." For example, "www.microsoft.com/ms.htm" is the homepage for Microsoft Corporation. The page is actually designated by "ms.htm," and the Web server on which the page resides is denoted by "www.microsoft.com." This is the default syntax for a URL. It actually points to a default Web page stored on a Web server with that Web address.

When the user clicks on a hypertext link, a series of actions take place behind the scenes. These actions are guided by the **hypertext transport protocol (HTTP)**. HTTP is a lightweight communication protocol that enables Web browsers and Web servers to converse with one another. The protocol has only seven commands. Two of these—GET and POST—make up the majority of requests issued by browsers. The GET command retrieves a resource (e.g., a Web page) specified by a URL. The POST command sends data (usually from a form) from the browser to the Web server.

WEB BROWSERS

The earliest Web browsers—Mosaic, Netscape 1.0, and Internet Explorer 1.0—were truly "thin" clients. Their primary function was to display Web documents containing text and simple graphics. Today, Microsoft's Internet Explorer (v. 6.0) has the lion's share of the browser market. Internet Explorer 6.0 is anything but thin, offering a suite of functions and features that are summarized in Exhibit A.4.

WEB SERVERS

In the computer world, the term *server* is often used to refer to a piece of hardware. However, a Web server is not a computer; it is a software program that runs on a computer. At last count, over 75 different Web servers were on the market. The primary function of all of these programs is to service HTTP requests. In addition, they also perform the following functions (Mudry 1995; Pffafenberger 1997):

- Provide access control, determining who can access particular directories or files on the Web server.
- Run scripts and external programs to either add functionality to the Web documents or provide real-time access to database and other dynamic data. This is done through various application programming interfaces such as CGI.
- Enable management and administration of both the server functions and the contents of the Web site (e.g., list all the links for a particular page at the site).
- Log user transactions. These transaction files provide data that can be statistically analyzed to determine the general character of the users (e.g., what browsers they are using) and the types of content that are of interest to them.

Although they have several functions in common, Web servers can be distinguished by:

- **Platform.** Some Web servers are designed solely for the Unix platform, others for Windows NT, and others for a variety of platforms.
- **Performance.** Web servers exhibit significant differences in processing efficiency, as well as the number of simultaneous requests they can handle and the speed with which they process those requests.
- **Security.** In addition to simple access control, some servers provide additional security services such as support for advanced authentication, access control by filtering the IP address of the person or program making a request, and support for encrypted data exchange between the client and server.
- **Commerce.** Some servers provide advanced services that support online selling and buying (e.g., shopping cart and catalog services). Although these advanced services can be provided with a stan-

EXHIBIT A.4 Browser Modules

Feature	Internet Explorer 6.0
Browser	Internet Explorer
Scripting support	JavaScript, VB Script
Active object support	Java, ActiveX
E-mail	Outlook Express
Web page authoring	FrontPage Express
Audio	Media Player
Streaming media	Media Player
Instant messaging	Microsoft's IM

dard Web server, they must be built from scratch by an application programmer rather than being provided "out of the box" by the server.

COMMERCIAL WEB SERVERS

Although there are dozens of Web servers on the market, two servers predominate—Apache and Microsoft's Internet Information Server. The following discussion provides a brief description of each.

The Apache server is free from apache.org. This server runs on a variety of hardware, including low-end PCs running the Linux and Windows operating systems. It also has a number of functions and features usually found with more expensive servers. It is also supported by a large number of third-party tools. A commercial version of the server, called Stronghold, is available from RedHat (redhat.com). Stronghold is a secure SSL Web server that provides full-strength, 128-bit encryption.

Microsoft Internet Information Server (IIS) is included (free of charge) with Windows NT, Windows 2000, or Windows XP. Like other Windows products, IIS is easy to install and administer. It also offers an application development environment, Active Server Pages (ASP), and an application programming interface (ISAPI) that makes it possible to easily develop robust, efficient applications. Like Apache, IIS can run on inexpensive PCs.

Since 1995, a company called Netcraft (netcraft.com) has been surveying Web servers connected to the "public" Internet in order to determine market share by vendor. This is done by physically polling all of the known Web sites with an HTTP request for the name of the server software. Since 1999, Apache has had between 50 to 65 percent market share, and Microsoft IIS has had between 20 and 35 percent. In October 2003, their respective shares were 65 percent and 25 percent (Netcraft 2003).

MULTIMEDIA DELIVERY

In addition to delivering Web pages with text and images, Web servers can also be used to download audio and video files of various formats (e.g., .mov, .avi, and .mpeg files) to hard disk. These files require a stand-alone player or browser add-in to hear and/or view them. Among the most popular multimedia players are RealNetwork's RealMedia Player, Microsoft's Windows Media Player, and Apple's QuickTime. Web servers can also be used to deliver audio and/or video in real time, assuming that the content is relatively small, that the quality of the transmission is not an issue, or that the content is not being broadcast live.

Streaming is a term used to refer to the delivery of content in real time. Streaming can be *on demand* or *live* (Mack 2002). Obviously, if the content is delivered on demand, then the content must exist ahead of time in a file. On-demand streaming is also called *HTTP streaming*. With on-demand streaming, if an end user clicks on a Web page link to an audio and/or video file, the file is progressively downloaded to the desktop of the end user. When enough of the file has been downloaded, the associated media player will begin playing the downloaded segment. If the media player finishes the downloaded segment before the next segment arrives, playback will be paused until the next segment arrives.

streaming
The delivery of content in real time; consists of two types, *on demand* (HTTP streaming) and *live* (true streaming).

The streaming of live broadcasts is called *true streaming* (Viken 2000). True streaming is being used with online training, distance learning, live corporate broadcasts, video conferencing, sports shows, radio programs, TV programs, and other forms of live education and entertainment. The quality of the audio that is delivered with true streaming can range from voice quality to AM/FM radio quality to near-CD quality. In the same vein, the quality of true video streaming can range from a talking-head video delivered as a 160×120 pixel image at a rate of 1 to 10 frames per second to quarter-screen animation delivered as a 300×200 pixel image at 10 frames per second to full-screen, full-motion video delivered in a 640×480 pixel window at 20 to 30 frames per second. You can think of a pixel as a small dot on the screen.

The real challenge in delivering streaming media is the bandwidth problem. For example, 5 minutes of CD-quality audio requires about 50 megabytes of data. Given that 1 byte equals 8 bits, it would take hours to download the file with a 56K modem. Several techniques (Ellis 2000) are used to overcome the bandwidth problem:

- Compared to television shows, which are displayed in a 640×480 pixels image at 30 frames per second, streaming videos are usually displayed in small areas at lower frame rates.
- With video streams, sophisticated compression algorithms are used to analyze the data in each video frame and across many video frames to mathematically represent the video in the smallest amount of data possible.
- With audio streams, sampling rates are reduced, compression algorithms are applied, and sounds outside the range of human hearing are discarded.

Streams and files are compressed for a specific expected transmission rate. For instance, if end users are accessing the streams with a 56K modem, then the resulting compression will be greater (i.e., the file size will be smaller) and the quality will be lower (i.e., the frames per minute will be slower) than if they were accessing the streams with a cable modem.

codecs
The compression algorithms that are used to encode audio and video streams; short for *compression and decompression*.

User Datagram Protocol (UDP)
Transport protocol used in place of TCP by streaming servers.

Real-Time Protocol (RTP)
Streaming protocol that adds header information to the UDP packets, thus enabling the synchronized timing, sequencing, and decoding of the packets at the destination.

Real-Time Streaming Protocol (RTSP)
Streaming protocol that adds controls for stopping, pausing, rewinding, and fast-forwarding the media stream; it also provides security and enables usage measurement and rights management.

peer-to-peer (P2P)
Applications that use *direct* communications between computers (peers) to share resources, rather than relying on a centralized server as the conduit between client devices.

The compression algorithms that are used to encode audio and video streams are called **codecs** (short for *compression and decompression*). Special tools are used to perform the compression. With on-demand streaming, the audio and video files are stored in compressed form. With true streaming, the content is compressed on the fly. In both cases, the media player decompresses the content. Unfortunately, different media players work with different compressed formats. For instance, the RealMedia player requires the real media format (.rm), whereas Microsoft's Windows Media Player uses the Advanced Streaming Format (.asf). Both of these are proprietary formats. MPEG-4, an audio/video compression format that has been adopted by the International Standards Organization (ISO), is being promoted as an open streaming standard.

True streaming requires specialized streaming servers, such as RealNetworks' RealServer or Microsoft's Windows Media Server, to deliver the live content. Streaming servers use different communication protocols than regular Web servers. More specifically, they employ a transport protocol called **User Datagram Protocol (UDP)**, rather than TCP, along with two streaming protocols—**Real-Time Protocol (RTP)** and **Real-Time Streaming Protocol (RTSP)**. RTP adds header information to the UDP packets. This information is used to enable the synchronized timing, sequencing, and decoding of the packets at the destination. RTSP is an application protocol that adds controls for stopping, pausing, rewinding, and fast-forwarding the media stream. It also provides security and enables usage measurement and rights management so that content providers can control and charge for the usage of their media streams.

P2P APPLICATIONS

Most Internet and Web applications are built on a client/server model, with the server housing the data and hosting the application. Over the past couple of years, a new set of distributed applications has arisen. These applications use *direct* communications between computers to share resources—storage, computing cycles, content, and human presence—rather than relying on a centralized server as the conduit between client devices. In other words, the computers on the "edge" of the Internet are peers, hence the name **peer-to-peer (P2P)** applications.

For years, the entire Internet had one model of connectivity. Computers were assumed to be on and connected at all times, and thus were given permanent IP addresses. The DNS was established to track those addresses. The assumption was that addresses were stable, with few additions, deletions, or modifications. Then, around 1994, the Web appeared. To access the Web with a browser, a PC needed its own IP address to connect to the Internet. In this environment, computers entered and left the Internet at will.

To handle the dynamic nature of the Web and the sudden demand for connectivity, ISPs began assigning IP addresses dynamically, giving client PCs a new address each time they connected to the Web. Because there was no way to determine which computer had a particular address, these PCs were not given DNS entries and, as a consequence, could not host applications or data.

P2P changes all of this. Just like the Web, computers on a P2P network come and go in an unpredictable fashion and do not have fixed IP addresses. Unlike the Web, the computers in a P2P network operate outside the DNS. This enables the computers in a P2P network to act as a collection of equals with the power to host applications and data. This is what makes P2P different from other Internet applications.

If you want to know whether an application is P2P, you need to determine (1) whether connectivity is variable and temporary network addresses are the norm and (2) if the nodes at the edge of the network are autonomous (Shirky 2000). ICQ, an instant messaging application, was one of the first P2P applications. ICQ relies on its own protocol-specific addresses that have nothing to do with the DNS. In ICQ all of the (chat) clients are autonomous. Gnutella (gnutella.com), a well-known file distribution application, is also P2P because the addresses of its nodes bypass the DNS and the nodes control file transfers.

A wide variety of P2P applications exist. The O'Reilly Network (oreillynet.com) provides an up-to-date directory of existing P2P applications (openp2p.com). These applications can be divided into one of five categories (Berg 2001): access to information, instant messaging, collaboration, distributed computation, and business process automation.

ACCESS TO INFORMATION

P2P applications providing access to information make it possible for one computer to share files with another computer located somewhere on the Internet. Essentially, the Internet or intranet becomes one big disk drive whose files can be located and transported with the P2P application. In the business world, P2P is used to create "affinity communities" where interested parties can share a collection of files on key business matters (e.g., strategic documents, white papers, etc.). The files can not only be viewed, but also moved from one computer to another.

The earliest of the file sharing applications was Napster. After a long series of lawsuits by the Recording Industry Association of America (RIAA) and various media companies against Napster and

Bertelsmann (the German media company that attempted to purchase Napster), Napster finally declared bankruptcy and shut down its operations for good at the end of 2002. In its place, a whole host of file-sharing programs and services have sprung up. KaZaA's Media Desktop (kazaa.com), which is used to share files of all sorts (audio/music, document, image, playlist, software, and video), is purported to be one of the most downloaded software programs in history, with over 230 million downloads. Gnutella is another well-known file-sharing application. Technically, Gnutella is not an application. Instead, it is a networking protocol that defines the manner in which the computers on the Gnutella network communicate with one another in a decentralized fashion in order to share files. Software vendors such as Lime Wire LLC (limewire.com) and Xolox (xolox.com) have developed file-sharing applications that are compatible with the Gnutella protocol.

Although the application functionality is basically the same, there are two P2P file-sharing models (limewire.com/index.jsp/p2p). One model is based on a central server system that directs traffic among the nodes. This was the model used by Napster. The central server maintains a directory of shared files that exist on the PCs of registered users. The directory is updated when the PC connects to the server network. When a user requests a particular file, the server creates a list of matching files on the PCs that are currently connected. The user selects the file from the list, at which point a direct HTTP connection is made between the user's PC and the PC possessing the file. The file is transferred directly between the PCs. The main advantage of this model is that the index maintained by the central server is both comprehensive and efficient.

The second model is completely decentralized. With this model, each client contacts one or more other clients to link into the network. Each client serves as a search engine for its neighbors, passing search requests throughout the network one node at a time. This is the model used by Gnutella. With Gnutella, each computer on the network has a Gnutella "servant"—a program that combines server and client functionality. An end user employs the servant to connect their computer to another computer on the Gnutella network. In turn, that computer announces to all the computers to which it is connected that another computer has joined the network. In turn, those computers announce the presence of the newly connected computer to the computers to which they are connected, and so on. When an end user wants to search for a file, the request is sent to the computers to which the user's computer is directly connected. In turn, the request is passed on to the computers to which they are connected, and so on until a match is found. At that point the computer with the matching file will send the file information back through the connected computers to the computer making the request. The user can then employ the servant to download the file directly from the computer with the matching file. This is done through HTTP. Although not as efficient as the first model, this model is very robust because it does not depend on a central point of contact.

INSTANT MESSAGING

Since their inception, instant messaging (IM) programs such as ICQ, AOL's Instant Messenger (AIM), MSN Messenger, and Yahoo! Messenger have been a tremendous hit. These programs enable end users to send notes back and forth with other IM users, create chat rooms where they can converse with other interested parties, share Web links, look at images on other people's computers, and play sounds for other people. When we think of instant messaging, we tend to think of chatting with our family and friends. However, IM has also established a presence in the corporate world.

Like Napster, most IM products are based on a central server model and work in essentially the same way. The products consist of two parts—IM clients and an IM server. The communication protocol that the clients use to converse with one another and with the server varies from one vendor to the next. For instance, AOL's IM uses a different protocol than MSN Messenger. This is why most IM products are unable to converse with one another.

When an end user opens an IM client, the client connects to the IM server. Once connected, the user logs into the server. After the server has verified the user's ID and password, the client sends the server its connection information, including its IP address and the port that client is using for messaging. Next, the server creates a temporary file that has the connection information along with a list of the end user's contacts (in the AOL terminology this is the "buddy list"). The server checks to see if any of these contacts are logged in. If any of the contacts are logged in, the server sends the connection information for those contacts to the end user's client. At the same time, it sends the client's connection information to the contacts' PCs.

When an end user clicks on an online contact, a messaging window opens. The end user then enters a message and clicks "Send." Because the IM client has the IP address and port number for the contact's computer, the message is sent directly to the contact's machine, bypassing the central server. The message that is sent appears in the contact's messaging window. The contact can then respond in a like manner. The conversation proceeds in this way until one of the participants closes the messaging window.

Eventually, when the end user goes off-line and exits from the IM client, the client sends a message to the server to terminate the session. At this point, the server will inform the PCs on the end user's contact list that the end user is no longer online. The temporary file containing the client connection information will be deleted.

COLLABORATION

Collaboration applications are the P2P version of a class of software applications that used to be called *groupware*. As the name implies, groupware was designed to support workgroup activities, such as the joint creation of a project document. In the same vein, these P2P applications are designed to support the collaborative activities of groups of individuals.

In reality, the applications within this category actually combine the features of the file-sharing applications with the functions of the instant messaging applications and provide support for various joint activities (e.g., conferencing). More specifically, these applications use a central server P2P model to provide the following types of capabilities: communications (instant messaging, chat, threaded discussions, content sharing); shared files, images, contacts, and virtually any other sort of data and information; and joint activities (real-time conferencing, white boarding, simultaneous browsing of documents or other files, and coediting of documents). One example of a P2P collaborative application is the Groove Network (groove.net). The Groove Network was designed by Ray Ozzie (the original designer of a well-known groupware application, Lotus Notes).

DISTRIBUTED COMPUTATION

By one very conservative estimate, the Internet has at least 10 billion MHz of PC processing power and 10 thousand terabytes of disk storage, assuming that each of the PCs only has a 100 MHz chip and a 100 MB hard drive (which is paltry by today's standards) (Cortese 2001). Much of this processing power and storage goes unused. Imagine if you could harness these unused resources to solve complex computational problems. This is what **P2P distributed computation** does. It uses P2P resource sharing to combine the idle processing cycles of computers on the network to form a *virtual computer* across which large computational jobs can be distributed.

P2P distributed computation
Computer architecture that uses P2P resource sharing to combine idle computer resources over a network, forming a *virtual computer* across which large computational jobs can be distributed.

One well-known distributed computation application is Seti@Home (setiathome.ssl.berkley.edu). This application uses more than 2 million computers on the Internet to analyze radio signals gathered from the Arecibo Observatory in Puerto Rico to search for extraterrestrial life. Another well-known example is Distributed.Net's (distributed.net) use of 100,000 PCs on the Internet to crack the 56-bit DES encryption algorithm. The algorithm was successfully cracked in July 2002. Currently, they are working on a new project to crack a message encrypted with the 72-bit RC5 cipher. Of course, distributed computation has been applied to less exotic domains, such as the financial service arena, which has used distributed P2P processing to solve complex financial models.

BUSINESS PROCESS AUTOMATION

Many organizational tasks involve the flow and processing of data and information across a network. For instance, take the budget approval process. Budgeting involves the allocation of resources within an organization to accomplish strategic aims. During the budgeting process, data are collected, bottom up, from a variety of people throughout an enterprise. The data that are submitted are usually reviewed to see if they fit with the strategic aims. The review process can involve several people. If the data are finally approved, then they move to the next step in the budgeting processes. At this stage, the data are aggregated with other submissions to arrive at a budget for an entire business unit or the whole company. If the data are rejected, they are returned to the person(s) who originally submitted them. At this point, the data are modified and submitted again for approval.

For most organizations, the approval process is done manually. Often, the data are e-mailed from one person to another for approval. If only a few people were involved, then this might suffice. However, in a large enterprise, budgeting can involve hundreds of people. Without an automated process to track and control the flow, data easily falls through the cracks. One way to automate these sorts of business processes is to use a "spoke and hub" architecture where the data flows from one client machine to another client machine via a centralized server. Typically, the data are moved from one client machine to the database, which is located on the server. The database contains information about which client should receive the data next. (In other words, rather than moving directly from one client machine to another, the data move from one client machine to the centralized server, and then on to the next machine.) The centralized server tracks the data flow process and sends information to the appropriate client.

An alternative architecture for automating business processes is to allow the nodes on the network to work directly with one another, passing data and information to the next node or nodes in the process. In a P2P application of this sort, software agents (see Appendix D) residing on each of the peer machines communicate with one another to determine the data flows, to search for other files and information if needed, and to prioritize tasks on the network.

IMPEDIMENTS TO P2P

Although P2P applications such as IM enjoy widespread use, some major impediments to the continued growth of P2P exist. The first problem is performance. In a client/server application, the bottleneck is the processing speed of the server. In P2P, the performance of the application depends on the speed of the various network connections and the individual computers on the network. If any of these connections or machines are slow, then the performance of the application can degrade. It is one thing to deal with a single server whose performance is slow. It is a much more difficult task to deal with network links and peer computers over which you have little control.

A second problem is security. For example, most IM applications send unencrypted text from one computer to another. This text can be easily captured and read by unauthorized parties. In the same vein, P2P file sharing and distributed processing applications usually bypass the firewall and let one machine control another. These applications are easy targets for hackers who can insert viruses or other rogue programs.

Third, in an enterprise environment, system administration can become a major hassle. It is very difficult to determine who has what version or who is authorized to use a particular application because many of the applications come from the outside. Finally, the P2P world has few standards. All of these applications rely on proprietary protocols. Although various standards bodies are at work (e.g., the Internet Engineering Task Force has proposed the Instant Messaging Presence Protocol), none of the protocols they might release are likely to impact P2P in the near future.

KEY TERMS

REFERENCES

Beardsley, S. "Making Sense of Broadband." *The McKinsey Quarterly*, **mckinseyquarterly.com**, 2003 no. 2 (accessed July 2003).

Berg, A. "P2P, or Not P2P?" *Information Security*, **infosecuritymag.techtarget.com/articles/february01/cover.shtml**, February 2001 (accessed July 2003).

Cortese, A. "Peer to Peer." *BusinessWeek Online*. **businessweek.com**, March 23, 2001 (accessed July 2003).

Ellis, R. "How to Stream Your Media Files." **washington.edu/computer/windows/issue24/file.htm**, Winter 2000 (accessed July 2003).

Harris, J. *Blindsided.* Oxford, UK: Capstone Publishing Ltd., 2002.

ISC. "Internet Domain Survey: January 2003." **isc.org/ds/www-200301/index.html**, January 2003 (accessed July 2003).

Loshin, P. *Extranet Design and Implementation.* San Francisco: Sybex Network Press, 1997, p. 374.

Mack, S. *Streaming Media.* New York: Wiley, 2002.

Mudry, R. *Serving the Web.* Scottsdale, AZ: Coriolis Group Books, 1995.

Netcraft.com. "Market Share for Top Servers Across All Domains, August 1995–October 2003." **netcraft.com**, **news.netcraft.com/archives/web_server_survey.html** (accessed October 2003).

Pffafenberger, B. *Building a Strategic Internet*. Foster City, CA: IDG Books, 1997.

Shirky, C. "What Is P2P . . . and What Isn't." **oreillynet.com**, November 2000 (accessed July 2003).

Treese, G., and L. Stewart. *Designing Systems for Internet Commerce.* Reading, MA: Addison-Wesley, 1998.

Viken, A. "Streaming: Past, Present, and Future." Master's thesis, Royal Institute of Technology: Stockholm, Sweden, 2000.

ADDITIONAL READINGS

Barkai, D. *Peer-to-Peer Computing: Technologies for Sharing and Collaborating on the Net.* Santa Clara, CA: Intel Press, 2002.

Edwards, J. "Not Just for Music Anymore." *Magazine*, **cio.com/archive/030101/et.html**, March 2001 (accessed July 2003).

Shirky, C., et al. *The Emergent P2P Platform of Presence, Identity, and Edge Resources*. Sebastopol, CA: O'Reilly & Associates, 2001.