Web Technology

The methods by which computers communicate with each other through the use of markup languages and multimedia packages are known as **web technology**. Web technologies relate the interface between web servers and their clients. This information includes markup languages, programming interfaces and languages, and standards for document identification and display.

History of World Wide Web

The World Wide Web allows computer users to locate and view multimedia-based documents (i.e., documents with text, graphics, animations, audios or videos) on almost any subject. Even though the Internet was developed more than three decades ago, the introduction of the World Wide Web is a relatively recent event. In **1990, Tim Berners-Lee** of CERN (the European Laboratory for Particle Physics) developed the World Wide Web and several communication protocols that form the backbone of the Web. The Internet and the World Wide Web surely will be listed among the most important and profound creations of humankind. In the past, most computer applications executed on "stand-alone" computers (i.e., computers that were not connected to one another). Today's applications can be written to communicate with hundreds of millions of computers. The Internet mixes computing and communications technologies. It makes our work easier. It makes information instantly and conveniently accessible worldwide. Individuals and small businesses can receive worldwide exposure on the Internet. It is changing the nature of the way business is done. People can search for the best prices on virtually any product or service. Special-interest communities can stay in touch with one another and researchers can learn of scientific and academic breakthroughs worldwide.

World Wide Web Consortium (W3C)

In October 1994, Tim Berners-Lee founded an organization—called the World Wide Web Consortium (W3C)—devoted to developing non-proprietary, interoperable technologies for the World Wide Web. One of the W3C's primary goals is to make the Web universally accessible—regardless of disability, language or culture. The W3C is also a standardization organization. Web technologies standardized by the W3C are called Recommendations. W3C Recommendations include the Extensible HyperText Markup Language (XHTML), Cascading Style Sheets (CSS), HyperText Markup Language (HTML; now considered a "legacy" technology) and the Extensible Markup Language (XML). A recommendation is not an actual software product, but a document that specifies a technology's role, syntax, rules, etc.

Before becoming a W3C Recommentation, a document passes through three phases:
Working Draft—which, as its name implies, specifies an evolving draft.
Candidate Recommendation—a stable version of the document that industry may begin implementing.

Proposed Recommendation—a Candidate Recommendation that is considered mature (i.e., has been implemented and tested over a period of time) and is ready to be considered for W3C Recommendation status.

History of the Internet

In the late 1960s, one of the authors (HMD) was a graduate student at MIT. His research at MIT's Project Mac (now the Laboratory for Computer Science—the home of the World Wide Web Consortium) was funded by ARPA—the Advanced Research Projects Agency of the Department of Defense. ARPA sponsored a conference at which several dozen ARPA-funded graduate students were brought together at the University of Illinois at Urbana-Champaign to meet and share ideas. During this conference, ARPA rolled out the blueprints for networking the main computer systems of about a dozen ARPA-funded universities and research institutions. They were to be connected with communications lines operating at a then-stunning 56Kbps (i.e., 56,000 bits per second)—this at a time when most people (of the few who could) were connecting over telephone lines to computers at a rate of 110 bits per second. HMD vividly recalls the excitement at that conference. Researchers at Harvard talked about communicating with the Univac 1108 "supercomputer" at the University of Utah to handle calculations related to their computer graphics research. Many other intriguing possibilities were raised. Academic research was on the verge of taking a giant leap forward. Shortly after this conference, ARPA proceeded to implement the ARPAnet, the grandparent of today's Internet. Things worked out differently from what was originally planned. Rather than the primary benefit of researchers sharing each other's computers, it rapidly became clear that enabling the researchers to communicate quickly and easily among themselves via what became known as electronic mail (e-mail, for short) was the key benefit of the ARPAnet. This is true even today on the Internet, as e-mail facilitates communications of all kinds among millions of people worldwide. One of the primary goals for ARPAnet was to allow multiple users to send and receive information simultaneously over the same communications paths (such as phone lines). The network operated with a technique called packet-switching, in which digital data was sent in small packages called packets. The packets contained data address, error control and sequencing information. The address information allowed packets to be routed to their destinations. The sequencing information helped reassemble the packets (which, because of complex routing mechanisms, could actually arrive out of order) into their original order for presentation to the recipient. Packets from different senders were intermixed on the same lines. This packet-switching technique greatly reduced transmission costs compared with the cost of dedicated communications lines. The network was designed to operate without centralized control. If a portion of the network should fail, the remaining working portions would still route packets from senders to receivers over alternate paths. The protocols for communicating over the ARPAnet became known as TCP—the Transmission Control Protocol. TCP ensured that messages were properly routed from sender to receiver and that

those messages arrived intact. As the Internet evolved, organizations worldwide were implementing their own networks for both intraorganization (i.e., within the organization) and interorganization (i.e., between organizations) communications. A wide variety of networking hardware and software appeared. One challenge was to get these different networks to communicate. ARPA accomplished this with the development of IP—the Internetworking Protocol, truly creating a "network of networks," the current architecture of the Internet. The combined set of protocols is now commonly called TCP/IP. Initially, Internet use was limited to universities and research institutions; then the military began using the Internet. Eventually, the government decided to allow access to the Internet for commercial purposes. Initially, there was resentment among the research and military communities—these groups were concerned that response times would become poor as "the Net" became saturated with users. In fact, the exact opposite has occurred. Businesses rapidly realized that they could tune their operations and offer new and better services to their clients, so they started spending vasts amounts of money to develop and enhance the Internet. This generated fierce competition among the communications carriers and hardware and software suppliers to meet this demand. The result is that bandwidth (i.e., the information carrying capacity) on the Internet has increased tremendously and costs have decreased significantly. It is widely believed that the Internet has played a significant role in the economic prosperity that the United States and many other industrialized nations have enjoyed recently and are likely to enjoy for many years.

Types of Accounts

To connect to the Internet, you can use one of several types of accounts: PPP and SLIP accounts, UNIX shell accounts, or online services.

Internet PPP and SLIP Accounts

A Point-to-Point Protocol (PPP) or Serial Line Internet Protocol (SLIP) account is an Internet account that uses the PPP or SLIP communications protocol, respectively. These are the most popular accounts, because the most popular software—Internet Explorer, Netscape Navigator, Eudora, and other programs—are designed to work with PPP and SLIP accounts. PPP is a more modern communications protocol than SLIP, so choose PPP if you have a choice when opening an account (almost all Internet providers offer PPP). Occasionally, you may run into a compressed SLIP (CSLIP) account, which is a more efficient version of SLIP, but still isn't as good as PPP. This book refers to PPP, CSLIP, and SLIP accounts as PPP accounts or Internet accounts.

An Internet service provider (ISP) is an organization that provides dial-in Internet accounts, usually PPP, CSLIP, or SLIP accounts, but sometimes UNIX shell accounts. Thousands of ISPs exist in the U.S., including dozens of ISPs with access phone numbers throughout the country, and many with phone numbers in limited regions. For example, AT&T WorldNet has access phone numbers in all major U.S. cities, whereas SoVerNet has

phone numbers only in Vermont and surrounding states, but provides local access from many towns that AT&T WorldNet doesn't cover.

To use a PPP account, you need a PPP-compatible communications program, such as Windows 98's Dial-Up Networking program (described in the section "TCP/IP and Connection Software," later in this chapter). This program dials the phone by using your modem (or connects using a higher-speed equivalent, as described in Chapter 3), connects to your ISP, logs in to your account by using your user name and password, and then establishes a PPP connection, thus connecting your computer to the Internet. While connected, you can use a variety of programs to read your e-mail, browse the Web, and access other information from the Internet. When you are done, you use Dial-Up Networking to disconnect from your Internet account. (See Chapter 2 for more information on how to set up a connection to a PPP Internet account.)

UNIX Shell Accounts

Before the advent of PPP and SLIP accounts, most Internet accounts were text-only UNIX shell accounts, and these accounts are still available from some ISPs. You run a terminal-emulation program (a program that pretends that your PC is a computer terminal) on your PC to connect to an Internet host computer. Most Internet hosts run UNIX, a powerful but frequently confusing operating system, and you have to type UNIX commands to use a UNIX shell account. To send and receive e-mail or browse the Web, you run text-only programs, such as Pine (the most popular UNIX e-mail program) and Lynx (the most widely used UNIX Web browser). When you use a UNIX shell account, you don't see graphics or use a mouse, and you can't easily store information on your own computer. Some providers give you both a PPP account and a UNIX shell account; you use the PPP account for your regular Internet work, and the UNIX shell account only when you need to change your account's password. For information about using UNIX shell accounts, see Chapter 40.

Online Services

An online service is a commercial service that enables you to connect to and access its proprietary information system. Most online services also provide an Internet connection, e-mail, the World Wide Web, and, sometimes, other Internet services. Online services usually require special programs to connect to and use your account.

The three most popular online services are the following:

- America Online (AOL) The world's most popular online service, with a wide range of AOL-only features. To connect to AOL, read AOL e-mail, browse the Web, and access other AOL services, you use AOL's proprietary program: the latest version is AOL 4.
- CompuServe (CIS) One of the oldest online services, with an excellent selection of proprietary technical- and business-oriented discussion groups. CompuServe was purchased by America Online, so the two services may merge. CompuServe has access phone numbers in dozens of countries. To connect to CompuServe and access

its services, you use CompuServe's proprietary program: the latest version is CompuServe 4.

■ Microsoft Network (MSN) Microsoft's online service. You connect to MSN by using Dial-Up Networking, send and receive e-mail by using Outlook or Outlook Express, and browse the Web by using Internet Explorer.

Web Servers

Web servers are computers that have specific software that allows them to accept requests from **client** computers and return responses to those requests. That is, a specialized software called a web server responds to client (e.g., Web browser) requests by providing resources (e.g., HTML documents). For example, when users enter a Uniform Resource Locator (URL) address, such as www.deitel.com, into a Web browser, they are requesting a specific document from a Web server. The Web server maps the URL to a file on the server (or to a file on the server's network) and returns the requested document to the client. During this interaction, the Web server and the client communicate using the platform-independent **HyperText Transfer Protocol (HTTP)**, a protocol for transferring requests and files over the Internet (i.e., between Web servers and Web browsers).

Requesting documents from a Web server

HTTP Request Types: The two most common HTTP request types (request methods) are **GET** and **POST**. These request types retrieve and send client form data to a Web server.

System Architecture

A Web server is part of a multi-tier application, sometimes referred to as an n-tier application. Multi-tier applications divide functionality into separate tiers (i.e., logical groupings of functionality). Tiers can be located on the same computer or on separate computers. The figure 1 presents the basic structure of a three-tier application.

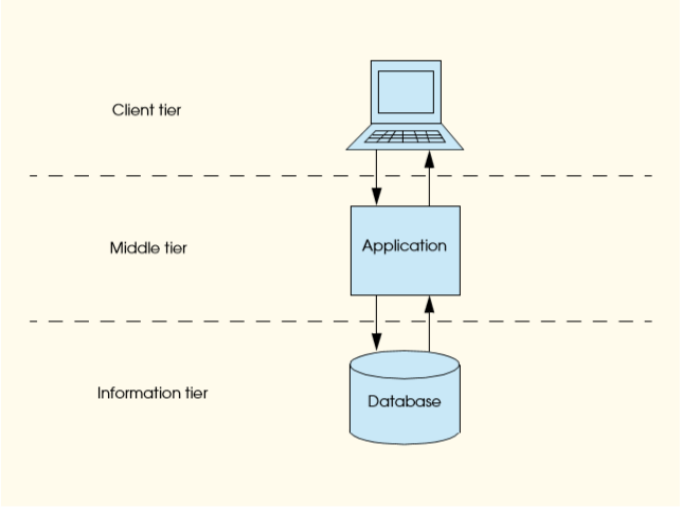

Figure 1 Three tier application model

The information tier (also called the data tier or the bottom tier) maintains data for the application. This tier typically stores data in a relational database management system (RDBMS).

The middle tier implements business logic and presentation logic to control interactions between application clients and application data. The middle tier acts as an intermediary between data in the information tier and the application clients. The middle-tier controller logic processes client requests from the top tier (e.g., a request to view a product catalog) and retrieves data from the database. The middle-tier presentation logic then processes data from the information tier and presents the content to the client. Business logic in the middle tier enforces business rules and ensures that data are reliable before updating the database or presenting data to a user. Business rules dictate how clients can and cannot access application data and how applications process data. The middle tier also implements the application's presentation logic. Web applications typically present information to clients as HTML documents.

The client tier, or top tier, is the application's user interface. Users interact directly with the application through the user interface. The client interacts with the middle tier to make requests and to retrieve data from the information tier. The client then displays the data retrieved from the middle tier to the user.

Client-Side Scripting versus Server-Side Scripting

Client-side scripting validates user input, accesses the browser and enhances web pages with ActiveX controls, Dynamic HTML and Java applets (i.e., client-side Java programs that execute in a browser). Client-side validation reduces the number of requests that need to be passed to the server. Interactivity allows users to make decisions, click buttons, play games, etc.— making a web site experience more interesting. ActiveX controls, Dynamic HTML and Java applets enhance a web page's functionality. Client-side scripts can access the browser, use features specific to that browser and manipulate browser documents. Client-side scripting does have limitations, such as browser dependency; the browser or scripting host must support the scripting language. Another limitation is that client-side scripts are viewable (e.g., by using the View menu's Source command in Internet Explorer) to the client. Some Web developers do not advocate this because users potentially can view proprietary scripting code. Sensitive information, such as passwords, should not be stored or validated on the client.

Programmers have greater flexibility when using server-side scripts. Scripts executed on the server usually generate custom responses for clients. For example, a client might connect to an airline's Web server and request a list of all flights from Boston to San Antonio between September 19th and November 5th. The server queries the database, dynamically generates HTML content containing the flight list and sends the HTML to the client. This

technology allows clients to obtain the most current flight information from the database by connecting to an airline's Web server.

Server-side scripting languages have a wider range of programmatic capabilities than their client-side equivalents. For example, server-side scripts can access the server's file directory structure, whereas client-side scripts cannot access the client's file directory. Server-side scripts also have access to server-side software that extends server functionality. These pieces of software are called ActiveX components for Microsoft Web servers and modules for Apache Web servers. Components and modules range from programming language support to counting the number of Web page hits.

Web Application

- A Web application is one that can be used by accessing a Web server through the Internet or an Intranet.
- They are popular due to the ease of use of Web browsers as "thin" clients.
- Another important reason for their popularity is the ability to update and maintain Web applications without distributing and installing software in thousands of potential clients.

Different Actors in Web application

There are different actors which comes into picture of web application.

- End User
- Web Client (Browser Application)
- Server (Web Server / Application Server)

User : It's the end user who actually raise the request for a page on Server or submit a form.

Web Client : Browsers(IE, Firefox..) are software that knows how to communicate with servers using HTTP protocol and render the page received from server in the form of HTML.

Server : Servers are kind of software which handle the user request and search for the content requested by end User and return the appropriate result (Requested Page or Error Code).

Server can be of two type

- web server
- App Server

What is HTML

HTML(Hyper Text Markup Language) is a language used by servers to send the response to Browser's so that they(Browsers) can render the page properly on user side.

What is HTTP Protocol

HTTP(Hyper Text Transfer Protocol) is protocol used by client(Browser) and server(Web Server) to communicate. HTTP has web specific featured that runs on TCP/IP protocol. Structure of HTTP conversation can be divided in two types, **HTTP Request** or **HTTP Response**.

HTTP Request

Whenever end User clicks any link or submit any form browser send HTTP Request to Server so that server can process it and provide the HTTP Response. HTTP Request's are of different kind. HTTP has several methods which inform the server about the kind of Request is being made by browser. Methods in HTTP Protocol are GET, POST, PUT, DELETE, HEAD, TRACE, OPTION, CONNECT. GET and POST are the method which is used mostly in web environment.

GET method is the default method which will be used by Browser. GET method is used by browser to ask (Not submitting any data) for some kind of resources on server (HTML, Image, PDF etc..). **POST method** is used by browser to send some information to server for processing.

Web Server

A Web server is a program that uses HTTP (Hypertext Transfer Protocol) to serve the files that form Web pages to users, in response to their requests, which are forwarded by their computers' HTTP clients. Dedicated computers and appliances may be referred to as Web servers as well.

The process is an example of the client/server model. All computers that host Web sites must have Web server programs. Leading Web servers include Apache, Microsoft's Internet Information Server (IIS) and nginx (Engine X) from NGNIX. Other Web servers include Novell's NetWare server, Google Web Server (GWS) and IBM's family of Domino servers.

What is Web Server and how it works?

Figure 2. Client - Server

Web server is a program that uses HTTP to serve files that create web pages to users in response to their requests, which are forwarded by their computers.

Any server that delivers an XML document to another device can be a Web server. A better definition might be that a Web server is an Internet server that responds to HTTP requests to deliver content and services.

Always a web server is connected to the internet. Every Web server that connects to the Internet will be provided with a unique address which was arranged with a series of four numbers between 0 and 255 separated by periods.

Also, web server enables the hosting providers to manage multiple domains(users) on a single server.

A web host is a company that leases out space on a cluster of servers to empower people to serve their own content & webpages.

Technologies

In order to better understand the underlying technology of applications developed for the Web, some additional concepts should be explained:

Different languages use different technologies to generate dynamic pages.

The CGI technology was developed first, followed by Web Classes. Currently, the technologies used include: CGI (for the Ruby generator), servlets and asp.net (for the Java and .NET generators, respectively).

Servlets

The word servlet is derived from another word, “applet”, which is used to refer to small programs written in Java to be executed in a Web browser. In comparison, a “servlet” is a program executed in a Web Server. The most common use of servlets is to generate Web pages dynamically based on the parameters of the request sent by the Web browser.

ASP.NET

Another alternative offered by Microsoft for developing Web applications. This technology was developed for the new .NET platform and it is an evolution of ASP or Active Server Pages. Again, in this case, they are applications that are executed on the server and send the resulting HTML code to the client. The programs in the server are .dlls invoked with .aspx extension.

JavaScript

Web applications also use JavaScript code to provide increased functionalities. JavaScript is an interpreted language to be used in Web pages, and its syntax is similar to that of the Java language. It has become a full-featured programming language that brings in the browser’s computing power to improve the usability of Web applications with advanced techniques, such as Ajax.

Ajax

Ajax is not a technology, but an architecture. It stands for Asynchronous JavaScript XML.
Features:
Based on presentation standards using XHTML and CSS;
Dynamic display and interaction using Document Object Model;
Data exchange and management using XML and XSLT;
Asynchronous data retrieval using XML Http Request; and JavaScript to bring these technologies together.

Different Types of Web Servers

There are different types of web servers available in open market. The most popular web servers are Apache, IIS, Nginx and LiteSpeed.

Apache Web Server

Apache is the most popular web server in the world developed by the Apache Software Foundation. Apache is an open source software and can be installed on almost all operating systems including Linux, Unix, Windows, FreeBSD, Mac OS X and more. About 60% of machines run on Apache Web Server.

An Apache server can be customized easily as it contains a modular structure. It is also an open source which means that you can add your own modules to the server when to require and make modifications that suit your specific needs.

It is more stable than any other web servers and is easier to solve administrative issues. It can be install on multiple platforms successfully.

Recent Apache releases provide you the feasibility of handling more requests when you compare to its earlier versions.

IIS Web Server

IIS is a Microsoft product. IIS server has all the features just like Apache. But it is not an open source and more over personal modules cannot be added easily and modification becomes a little difficult job.

Microsoft developed, maintains it, thus works with all the Windows operating system platforms. Also, they had good customer support if it had any issues.

Nginx Web Server

Nginx is another free open source web server, it includes IMAP/POP3 proxy server. Nginx is known for its high performance, stability, simple configuration and low resource usage.

This web server doesn't use threads to handle requests rather a much more scalable event-driven architecture which uses small and predictable amounts of memory under load. It is getting popular in the recent times and it is hosting about 7.5% of all domains worldwide.

LightSpeed Web Server

LiteSpeed (LSWS) is a high-performance Apache drop-in replacement. LSWS is the 4th most popular web server on the internet and it is a commercial web server. Upgrading your web server to LiteSpeed will improve your performance and lower operating costs.

This is compatible with most common Apache features, including mod_rewrite, .htaccess, and mod_security. LSWS can load Apache configuration files directly and works as a drop-in replacement for Apache with hosting control panels — replacing Apache in less than 15 minutes with zero downtime.

Unlike other front-end proxy solutions, LSWS replaces all Apache functions, simplifying use and making the transition from Apache smooth and easy. Most of the hosting companies were using LSWS in recent times.

References

1. Harvey M. Deitel, Paul J. Deitel, Tem R. Nieto, "Internet & World Wide Web How to Program", 4th edition, Prentice Hall
2. Margaret Levine Young , "Internet: The Complete Reference", Millennium Edition, McGraw-Hill Publications
3. https://wiki.genexus.com
4. http://javabeginnerstutorial.com/
5. https://www.tutorialspoint.com/