

# Introduction to Internet Programming

**Internet Programming I: Chapter 1**



ADDIS ABABA SCIENCE AND TECHNOLOGY UNIVERSITY  
**Department of Software Engineering**

# Internet programming



- Internet programming is Internet-based application programming (Internet applications)
- It is programming to:
  - Access, Control and deliver data across the Internet
  - Enable functionality distribution across the Internet
- Internet programming includes broad variety of technologies spanning diverse areas such as:
  - Protocols for communication networks
  - Interfacing to databases
  - Programming of graphical user interfaces, or structuring information

# Internet



- Sometimes called the “the Net” is a global system of interconnected computer networks that use the standard Internet protocol suite (**TCP/IP**)
- It is a network of networks that consists of millions of private, public, academic, business, and government networks, of local to global scope
- That are linked by a broad array of electronic, wireless and optical networking technologies
- The Internet has no centralized governance in either technological implementation or policies for access and usage; each constituent network sets its own policies
- Only the overarching definitions of the two principal name spaces in the Internet, the **Internet Protocol Address Space** and the **Domain Name System**, are directed by a maintainer organization, the Internet Corporation for Assigned Names and Numbers (ICANN)
- The technical underpinning and standardization of the core protocols (IPv4 and IPv6) is an activity of the Internet Engineering Task Force (IETF)

# Uses of the Internet



- The Internet allows greater flexibility in working hours and location
- Educational material at all levels from pre-school to post-doctoral is available from websites
- The low cost and nearly instantaneous sharing of ideas, knowledge, and skills has made collaborative work dramatically easier, with the help of collaborative software
- CMS allow collaborating teams to work on shared sets of documents simultaneously without accidentally destroying each other's work
- The Internet allows computer users to remotely access other computers and information stores easily, wherever they may be, etc...

# Services of Internet

- World Wide Web
- Communication
  - IRC(Internet Relay Chat), Email, List Server (LISTSERV), VoIP, User Network (USENET), Telnet, Video Conferencing
- Information Retrieval Services
- File Transfer: Gopher, FTP (File Transfer Protocol), Archie
- Directory Services: DNS (Domain Number System), LDAP (Lightweight Directory Access Protocol)
- Automatic Network Address Configuration: DHCP
- Network Management Services: ping, traceroute
- Time Services: Network Time Protocol (NTP), The Simple Network Time Protocol (SNTP)
- Other services: Usenet, News Group, E-commerce
- **The most widely used part of the Internet is the World Wide Web (often abbreviated "WWW" or called "the Web")**

# World Wide Web (WWW)



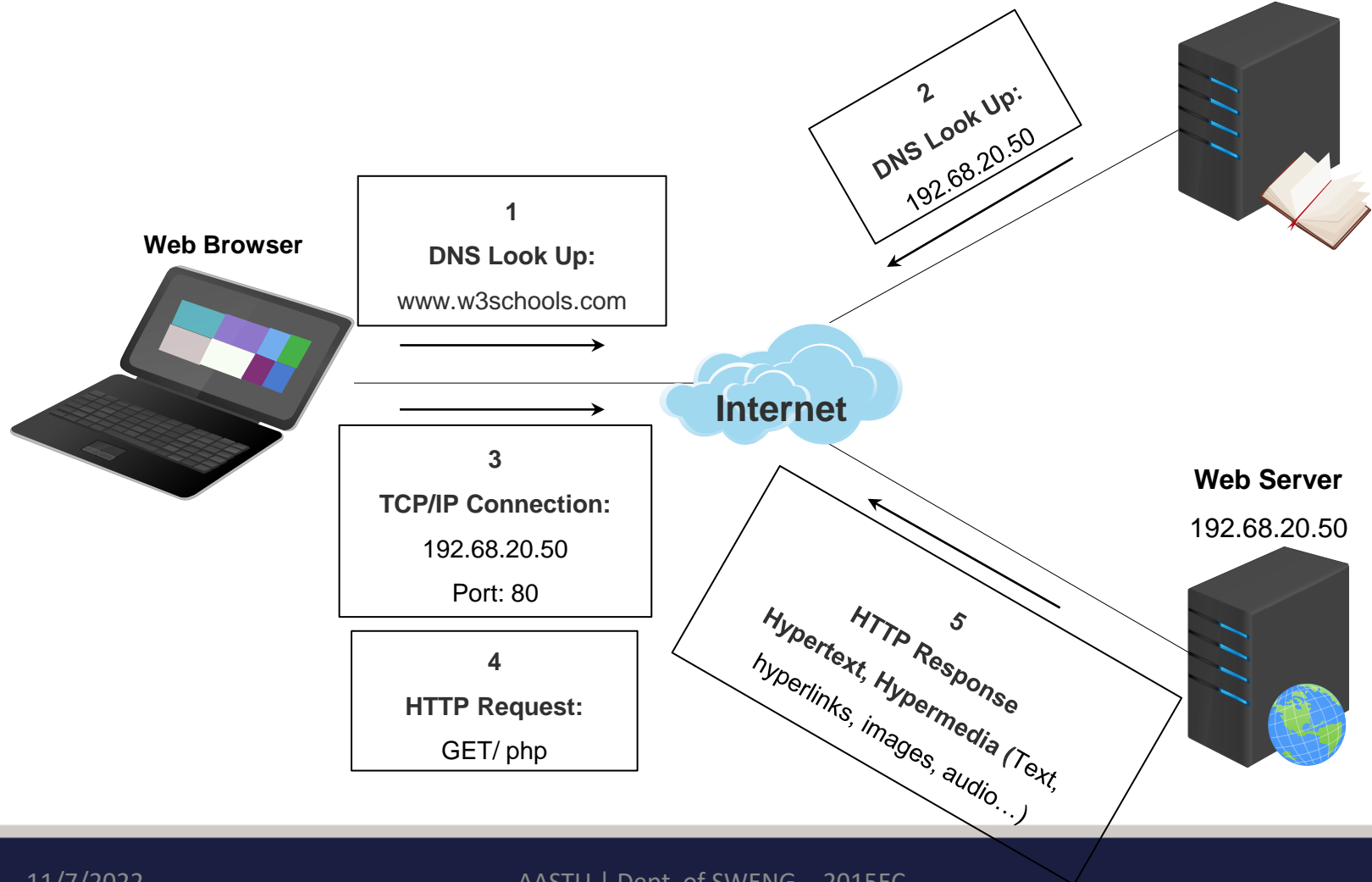
- The World Wide Web came into being in 1991, by developer Tim Berners-Lee and other team members
- The team created the protocol based on **hypertext** that makes it possible to connect content on the Web with **hyperlinks**
- World Wide Web (WWW) is a system of information resources accessed primarily through the Internet
- It is a set of Internet protocols and software that present information in a hypertext format
- Hypertext is a text which displayed on a computer display or on other electronic device with references (hyper link) to other text that the reader can immediately access
- WWW is a computer-based network of information resources that combines text and multimedia

# World Wide Web (WWW) cont'd



- WWW outstanding feature is **hypertext**, a method of instant cross-referencing
- In most Web sites, certain words or phrases appear in text of a different color than the rest; often this text is also underlined
- When you select one of these words or phrases, you will be transferred to the site or page that is relevant to this word or phrase
- Sometimes there are buttons, images, or portions of images that are "clickable" If you move the pointer over a spot on a Web site and the pointer changes into a hand, this indicates that you can click and be transferred to another site
- To view files on the Web, you need Web browsing software. You use this software to view different locations on the Web, which are known as Web pages
- A group of Web pages is a Web site. The first page of a Web site is often called the home page
- Just as each household in the world has a unique address, each Web page in the world has a unique Internet address, sometimes called a URL.
  - For example, the address of HTML course in the w3schools is <https://www.w3schools.com/html/>

# World Wide Web (WWW) cont'd





# World Wide Web (WWW) cont'd



- When you request a site the first thing your computer does is a DNS lookup this takes a domain name as input like “w3schools.com/html” and replies back with the matching computer's IP address
- Now armed with the IP address of the computer you want your web browser opens a TCP connection to a computer that's running a special piece of software called a web server
- The standard port number for web servers is port 80. At this point all your computer has done is connect to the web server at the address, e.g. w3schools
- The next step is to ask that web server for the courses, hypertext page, to do this it uses the Hypertext Transfer Protocol or HTTP
- The command get/html is sent as a raw ASCII text to the web server which then replies back with the web page hypertext we requested
- This is interpreted by your computer's web browser and render to your screen
- If the user follows a link to another page the computer just issues another get request and this goes on and on as you surf around the web site

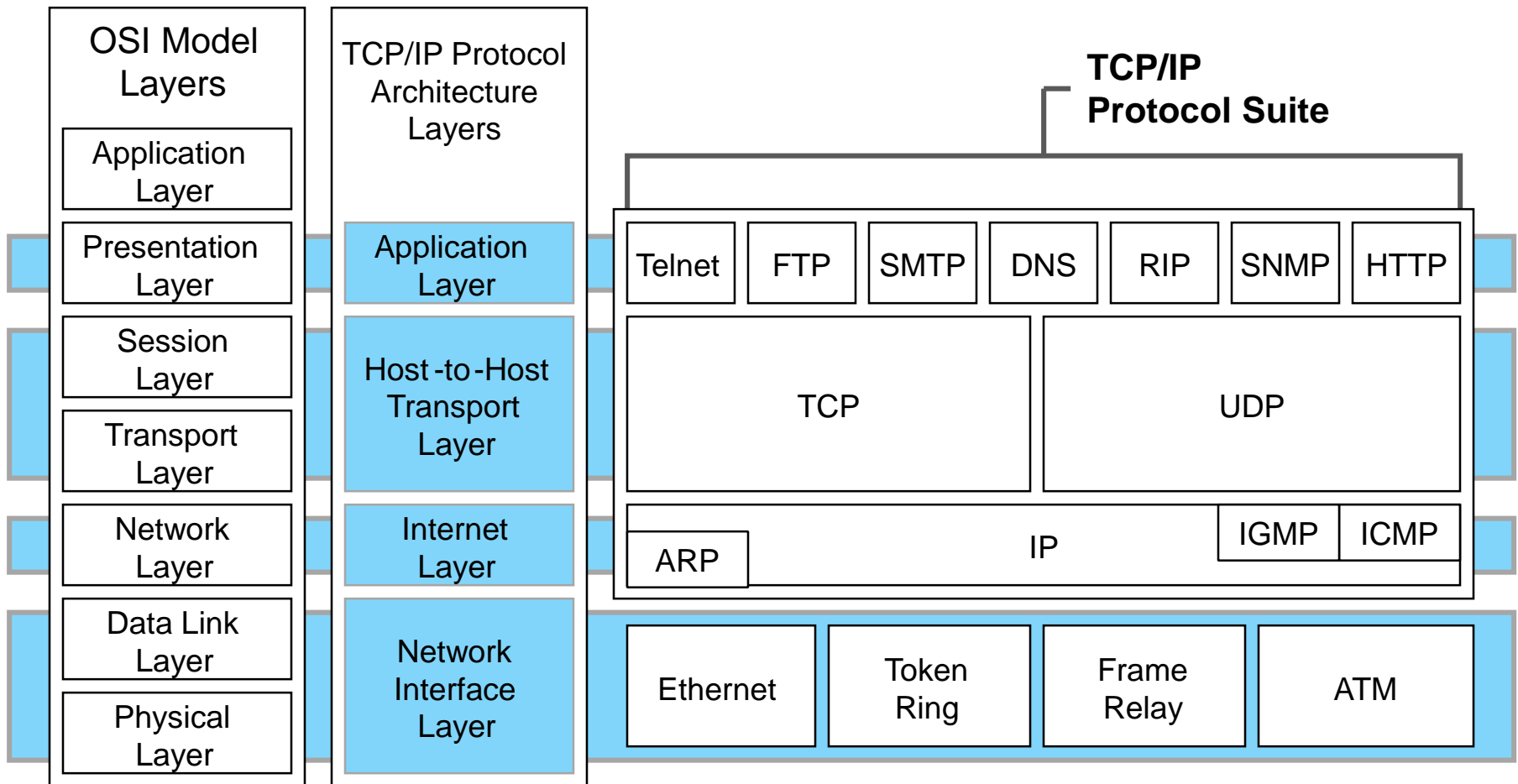
# Web Services and Protocols

# TCP/IP Protocol Suite



- TCP/IP is a set of protocols developed to allow cooperating computers to share resources across a network
- TCP stands for “Transmission Control Protocol”
- IP stands for “Internet Protocol”
- They are Transport layer and Network layer protocols respectively of the protocol suite
- The most well known network that adopted TCP/IP is Internet – the biggest WAN in the world

# TCP/IP Protocol Suite cont'd



# TCP/IP Protocol Suite cont'd



- Internet Protocol (IP)
  - Routable, connectionless datagram delivery
  - Specifies source and destination
  - Does not guarantee reliable delivery
  - Large message may be broken into many datagrams, not guaranteed to arrive in the order sent
- Transport Control Protocol (TCP)
  - Reliable stream transport service
  - Datagrams are delivered to the receiving application in the order sent
  - Error control is provided to improve reliability

# TCP/IP Application Layer



- Application layer protocols define the rules when implementing specific network applications
- Rely on the underlying layers to provide accurate and efficient data delivery
- Typical protocols include:
  - HTTP – Hypertext Transfer Protocol
    - For Web browsing
  - FTP – File Transfer Protocol
    - For file transfer
  - Telnet – Remote terminal protocol
    - For remote login on any other computer on the network
  - SMTP – Simple Mail Transfer Protocol
    - For mail transfer
- Servers on the Internet: HTTP, FTP, telnet, DNS, NNTP, Gopher...

# HTTP Protocol Client/Server, Request/Response architecture



- You request a Web page -> HTTP request
  - e.g. <http://www.aastu.edu.et/index.php>
- The Web server responds with data in the form of a Web page -> HTTP response
  - Web page is expressed as HTML
- Pages are identified as a Uniform Resource Locator (**URL**)
  - Protocol: http
  - Web server: [www.aastu.edu.et](http://www.aastu.edu.et)
  - Web page: index.asp
  - Can also provide parameters: ?name=Abebe
- HTTP is a stateless protocol, each HTTP request is independent of previous and subsequent requests
- Statelessness has a big impact on how scalable applications are designed

# HTTP Methods



- GET request-URI HTTP/1.1
  - Retrieve entity specified in request-URI as body of response message
- POST request-URI HTTP/1.1
  - Sends data in message body to the entity specified in request-URI
- PUT request-URI HTTP/1.1
  - Sends entity in message body to become newly created entity specified by request-URI
- HEAD request-URI HTTP/1.1
  - Same as GET except the server does not send specified entity in response message
- DELETE request-URI HTTP/1.1
  - Request to delete entity specified in request-URI
- TRACE request-URI HTTP/1.1
  - Request for each host node to rep



# HTTP Request



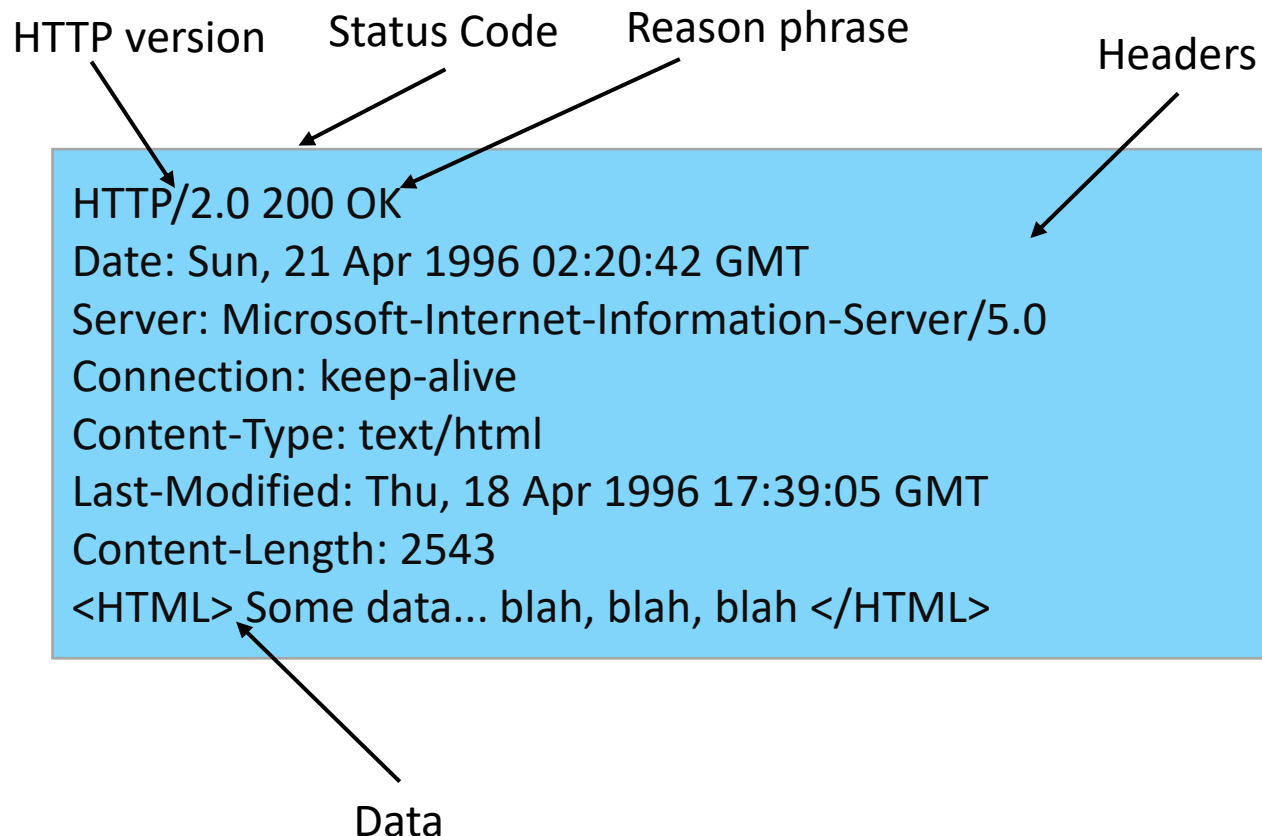
Method      File      HTTP version      Headers

```
GET /index.php HTTP/2.0
Accept: image/gif, image/x-bitmap, image/jpeg, */*
Accept-Language: en
User-Agent: Mozilla/1.22 (compatible; MSIE 2.0;
Windows 95)
Connection: Keep-Alive
If-Modified-Since: Sunday, 17-Apr-21 04:32:58 GMT
```

Blank line

Data – none for GET

# HTTP Response



# Status Codes



200 OK  
201 Created  
202 Accepted  
204 No Content  
301 Moved Permanently  
302 Moved Temporarily  
304 Not Modified  
400 Bad Request  
401 Unauthorized  
403 Forbidden  
404 Not Found  
500 Internal Server Error  
501 Not Implemented  
502 Bad Gateway  
503 Service Unavailable

## Classes:

**1xx:** Informational

- not used, reserved for future

**2xx:** Success

- action was successfully received, understood, and accepted

**3xx:** Redirection

- further action needed to complete request

**4xx:** Client Error

- request contains bad syntax or cannot be fulfilled

**5xx:** Server Error

- server failed to fulfill an apparently valid request

# Uniform Resource Locator (URL)

- Uniform Resource Locator (URL) is a Uniform Resource Identifier (URI)
  - Specifies where an identified resource is available and the mechanism for retrieving it
- Every server on the Internet has an IP number, a unique number consisting of 4 parts separated by dots
- The IP number is the server's address
  - 165.113.245.2
  - 128.143.22.55
- However, it is harder for people to remember numbers than to remember word combinations
- So, addresses are given "word-based" addresses called URLs
- The URL and the IP number are one and the same
- A URL looks like this:
  - <http://www.aastu.edu.et/>
  - <telnet://well.sf.ca.us>
  - <gopher://gopher.ed.gov/>

# Uniform Resource Locator (URL) cont'd



- The URL is divided into sections:

**<protocol>://<host>[:<port>][<path>][?<query>]**

- The first part of a URL defines the transport protocol.
  - http:// (Hypertext Transport Protocol) moves graphical, hypertext files
  - ftp:// (File Transfer Protocol) moves a file between 2 computers
  - gopher:// (Gopher client) moves text-based files
  - news: (News group reader) accesses a discussion group
  - telnet:// (Telnet client) allows remote login to another computer
- Here's an example:
  - http://www.vrml.k12.la.us/tltc/mainmenu.htm?lang=en
    - Protocol: http is the protocol
    - Host: www.vrml.k12.la.us is the server (Domain name )
    - Path: tltc/mainmenu.htm is the path, mainmenu.htm is the filename of the page on the site
    - Query: lang=en

# Domain Name System (DNS)



- Every computer has a network address
  - e.g. 158.132.161.99
- To access a computer, we need to specify its network address
- Human beings are weak in memorizing numbers
- We prefer computer name or domain name
  - e.g. hkpu10.polyu.edu.hk
- Need a machine on the Internet to convert name to number
- Designers of the Internet invented the Domain Name System (DNS)

# Domain Name System (DNS) cont'd



- DNS associates hostnames that humans can remember (like `aastu.edu.et`) with IP addresses that computers can remember (such as `197.156.73.161`)
- Every computer connected to the Internet should have access to a machine called a domain name server.
- Most domain name servers know the addresses of only the hosts on their local network, plus the addresses of a few domain name servers at other sites
- If a client asks for the address of a machine outside the local domain, then the local domain name server asks a domain name server at the remote location and relays the answer to the requester

# Web Browser



- A Web browser contains the basic software you need in order to find, retrieve, view, and send information over the Internet
- This includes software that lets you:
  - Send and receive electronic-mail (or e-mail) messages worldwide nearly instantaneously
  - Read messages from newsgroups (or forums) about thousands of topics in which users share information and opinions
  - Browse the World Wide Web (or Web) where you can find a rich variety of text, graphics, and interactive information
- The most popular browsers are Microsoft Internet Explorer, Google chrome and Netscape Navigator
- The appearance of a particular Web site may vary slightly depending on the browser you use



# Web Server



- A web server is a computer programs that delivers (serves) content, such as web pages, using the Hypertext Transfer Protocol (HTTP), over the World Wide Web
- You know when you are using HTTP because the website URL begins with http:// (for example, “http://www.google.com”)
- The term web server can also refer to the computer or virtual machine running the program
- Examples of the most popular web servers are:
  - Apache HTTP Server (also referred to as simply “Apache”)
  - Microsoft Internet Information Services (IIS)
  - Sun Java System Web Server

# Client Server Architecture

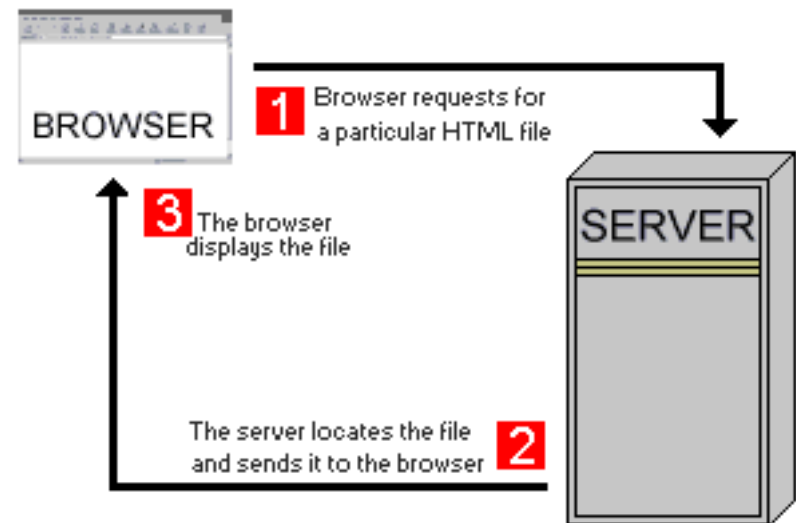


- The Internet revolves around the client-server architecture
- Your computer runs software called the client and it interacts with another software known as the server located at a remote computer
- The client is usually a browser such as Internet Explorer, Netscape Navigator or Mozilla
- Browsers interact with the server using a set of instructions called protocols
- These protocols help in the accurate transfer of data through requests from a browser and responses from the server
- There are many protocols available on the Internet
- The World Wide Web, which is a part of the Internet, brings all these protocols under one roof
- You can, thus, use HTTP, FTP, Telnet, SMTP, etc. from one platform - your web browser
- The web employs a connection-less protocol, which means that after every client-server interaction the connection between the two is lost

# Client-Server Architecture With Three Models

## Model 1: Static HTML pages

- The client (browser) requests for an HTML file stored on the remote machine through the server software
- The server locates this file and passes it to the client
- The client then displays this file on your machine
- In this case, the HTML page is static
- Static pages do not change until the developer modifies them

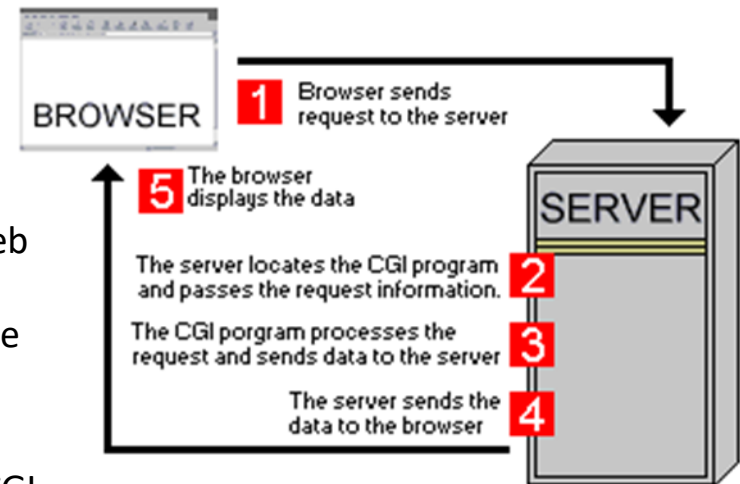


# Client-Server Architecture With Three Models



## Model 2: Common Gateway Interface (CGI) Scripts

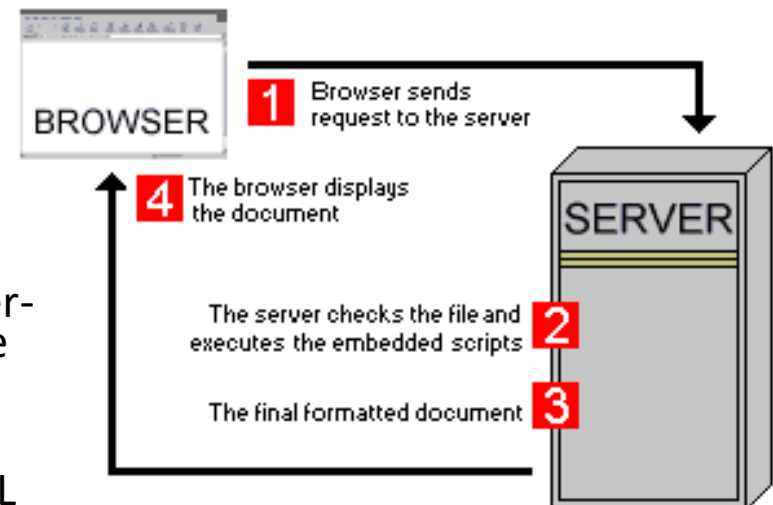
- The scenario is slightly different for CGI applications
- Here the server has to do more work since CGI programs consume the server machine's processing power
- Let us suppose you come across a searchable form on a web page that runs a CGI program
- Let us also suppose you type in the word 'computers' as the search query
- Your browser sends your request to the server
- The server checks the headers and locates the necessary CGI program and passes it the data from the request including your search query "computers"
- The CGI program processes this data and returns the results to the server
- The server then sends this formatted in HTML to your browser which in turn displays the HTML page
- Thus the CGI program generates a dynamic HTML page
- The contents of the dynamic page depend on the query passed to the CGI program



# Client-Server Architecture With Three Models cont'd

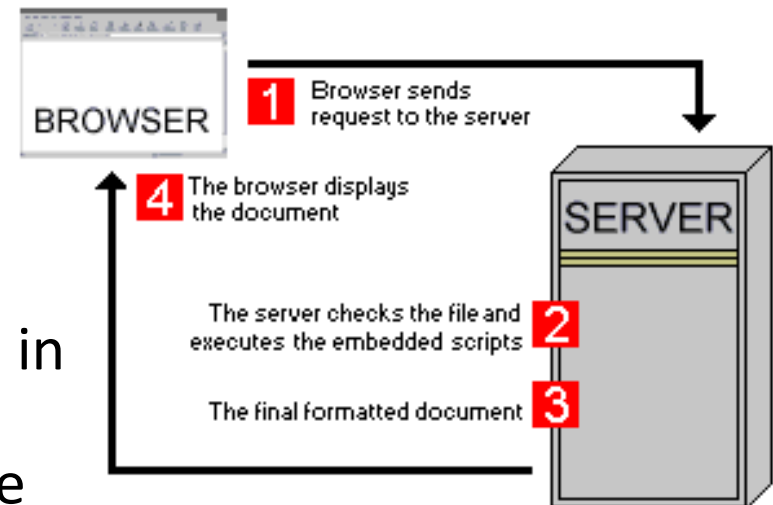
## Model 3: Server Side Scripting Technologies

- The third case also involves dynamic response generated by the use of server side technologies
- There are many server side technologies today ASP, PHP, JSP and SSI
- With these server technologies it has become easier to maintain Web pages especially helpful for a large web site
- The developer needs to embed the server-side language code inside the HTML page
- This code is passed to the appropriate interpreter which processes these instructions and generates the final HTML displayed by the browser
- Note, the embedded server-script code is not visible to the client (even if you check the source of the page) as the server sends ONLY the HTML code



# Client-Server Architecture With Three Models cont'd

- Let's look at PHP as an example
- A request sent for a PHP page from a client is passed to the PHP interpreter by the server along with various program variables
- The interpreter then processes the PHP code and generates a dynamic HTML output
- This is sent to the server which in turn redirects it to the client
- The browser is not aware of the functioning of the server
- It just receives the HTML code, which it appropriately formats and displays on your computer



# Server Side and Client Side Scripts

## **Client-Side Scripts:**

- program that runs on the client machine (browser) and deals with the user interface/display, and any other processing that can happen on client machine like reading/writing cookies
- Makes Interactive webpage with temporary and/or local storage
- Makes interactive web pages. Allows sending request for data to server and work as an interface between server and user
- The client-side scripts include: JavaScript, VBScript, HTML, CSS, AJAX

# Server Side and Client Side Scripts cont'd



## **Server-Side Scripts:**

- Program that runs on server dealing with the generation of content of web page
- Allows querying the database
- Operations over databases
- Access/Write a file on server
- Interact with other servers
- Structure web applications
- Process user input. For example if user input is a text in search box, run a search algorithm on data stored on server and send the results
- Some of the server side scripting languages include: PHP, C++, Java and JSP, Python, Ruby on Rails.



# Website Design and Development Processes and Good Practices



- Engineers should follow web development best practices to create products that effectively align with the industry standard and user expectations
- As a web engineer and developer, you might work on all kinds of web projects, some being relatively simple and others being more complex
- Regardless of the type or scale of the project you're working on, it's a good idea to get into some good habits and follow web development best practices.
- This minimize mistakes, cut down on wasted time, and ensure that all of your work is delivered to a good standard

# Website Design and Development Processes and Good Practices cont'd



- Some of good web development practices include:
- **Plan and Purpose** developers tend to work much more efficiently when they have a plan in mind and a purpose to aim for
  - Don't simply dive in and start coding without taking the time to plan out
    - what you're doing, why you're doing out and, how you're going to go about it
- **Accessibility**
  - Make sure that your sites and apps are comfortable for everyone to use
    - e.g. people worldwide having some sort of disability
  - Some ways to do this include adding ALT text to images so that screen readers can explain them and making designs friendly for those with color-blindness too
- **Keep the User Experience in**, thinking of how they'll interact with your pages or apps and making sure that you provide the best possible experiences for all
- **Keep Standards**
  - Rules, standards, and frameworks exist for a reason
  - Some rules, such as keeping global navigation menus and maps the same across a site or ensuring that hyperlink text size is appropriate, are not meant to be broken

# Website Design and Development Processes and Good Practices cont'd



- **Keep Code Concise and Useful**

- Complex codes can look more impressive, but they're usually much riskier than simpler alternatives, as they open the doors to more opportunities for unexpected bugs and issues
- Keep your code concise and make sure that every line actually matters

- **Compatibility Across Multiple Devices**

- It's important to ensure when developing web apps and pages to ensure that they can be accurately and appropriately displayed across screens of different sizes

- **Don't Mix HTML With CSS or JavaScript**

- Over this semester, you'll learn HTML, CSS, JavaScript, and so on
- While languages can cross over in certain ways, it's important to try and avoid combining them when they don't work well together
- HTML and CSS, for example, can get very messy when combined, and it isn't considered good practice to mix JavaScript in with your HTML documents either

- **Incremental updates**

- Web development is a rigorous process of improvement, and it thrives on incremental updates rather than implementing huge features at once
- Whether it's a minor change in fonts or introducing a new feature, incremental updates create fewer errors and bugs than full updates do

# Web Development Tools



- Web apps are expected to be highly responsive, sleek, and fast; the same can be said for web developers
- Professionals need to be highly adaptive, not resisting changes but instead taking advantage of new opportunities to build safer, faster, and more intuitive websites
- Web development is made much with the variety of tools
- Web development tools allow for greater efficiency and improved workflow, organization and anchoring the different skills needed to be a web developer
- Few web development tools include:
  - **Code/Text Editors:** whether you're writing in HTML, CSS, JavaScript, PHP, or any other applicable language, many code editors are available such as Notepad ++, Visual Studio Code and others
  - **Browser Developer Tools:** several web browsers have support for web developer tools that allow web designers and developers to look at the make-up of their pages. These are all tools that are built into the browser and do not require additional modules or configuration
  - **GitHub:** allows you back up your previous builds and code, find a community of online users happy to help answer questions, as well as helpful features for tracking code errors

# Web Development Tools cont'd



- **Local Development Tools:** XAMPP, MAMP, WAMP, DevKinsta
- **Web Design and Prototyping Tools:** are tools for prototyping your layouts and visuals
- **Frameworks:**
  - Bootstrap, Tailwind CSS, Bulma, Foundation, Material-UI, HTML Boilerplate, Materialize
  - JQuery, React, Vue, Express, Svelte, Angular, TypeScript
  - Laravel, Django
- **Package Managers:** Node Package Manager (npm), Yarn
- **API and Testing Tools:** Postman, HoppScotch, Testing Library
- **Collaboration Tools:** Jira, Asana, Basecamp, Taskade
- **Task Runners:** Grunt, Gulp
- **Containerization Tools:** Docker, LXD
- **Image Optimization Tools:**
- **Website Testing Tools:** Google Lighthouse, Responsively, Cypress
- **Stack Overflow and Search Engines**
- **Web Development References:** MDN Web Docs, DevDocs, CSS-Tricks, DEV Community, Can I use...
- These are some of the tools to mention. You might use one to many of these tools, one thing you should bear in mind is, as a developer, you should keep your eye open because there is always change

# Website Security Consideration



- Websites are always prone to security risks
- Cybercrime impacts your business by hacking your website/web based application
- Your system is then used for hacking assaults that install malicious software or malware on your users computer
- Hackers may also steal important customer data such as credit card information, destroy your business and propagate illegal content to your users
- So take the following security Considerations:
  - **Updated Software:** it is mandatory to keep you software updated. It plays vital role in keeping your website secure.
  - **SQL Injection:** it is an attempt by the hackers to manipulate your database. It is easy to insert rogue code into your query that can be used to manipulate your database such as change tables, get information or delete data.
  - **Cross Site Scripting (XSS):** it allows the attackers to inject client side script into web pages. Therefore, while creating a form, it is good to endure that you check the data being submitted and encode or strip out any HTML

# Website Security Consideration cont'd



- So take the following security Considerations(cont'd):
  - **Error Messages:** you need to be careful about how much information to be given in the error messages
    - For example, if the user fails to log in the error message should not let the user know which field is incorrect: username or password
  - **Validation of Data:** the validation should be performed on both server side and client side
  - **Passwords:** it is good to enforce password requirements such as of minimum of eight characters, including upper case, lower case and special character. It will help to protect user's information in long run
  - **Upload files:** the file uploaded by the user may contain a script that when executed on the server opens up your website
  - **SSL:** it is good practice to use SSL protocol while passing personal information between website and web server or database

# Search Engine Optimization



- Search Engine Optimization refers to set of activities that are performed to increase number of desirable visitors who come to your site via search engine
- These activities may include things you do to your site itself, such as:
  - Making changes to your text and HTML code
  - Formatting text or document to communicate directly to the search engine