# Internet Fundamentals & Introduction to Web Technologies

Course: IT (044615)

Lecture: 1

**Introduction**

**Dr. Ramez Hajislam**

# Brief Introduction to Internet

- ARPA net late 1960s – early 1970s
  - Network reliability, For ARPA funded research organizations
- BIT net, CS net - late 1970s & early 1980s
  - email and file transfer for other institutions
- NSF net - 1986
  - Originally for non-DOD funded places
  - Initially connected five supercomputer centers
  - By 1990, it had replaced ARPA net for non-military uses
  - Soon became the network for all (by the early 1990s)
- NSF net eventually became known as the Internet

# What the Internet is?

- A world-wide network of computer networks.
- At the lowest level, since 1982, all connections use TCP/IP
- TCP/IP hides the differences among devices connected to the Internet

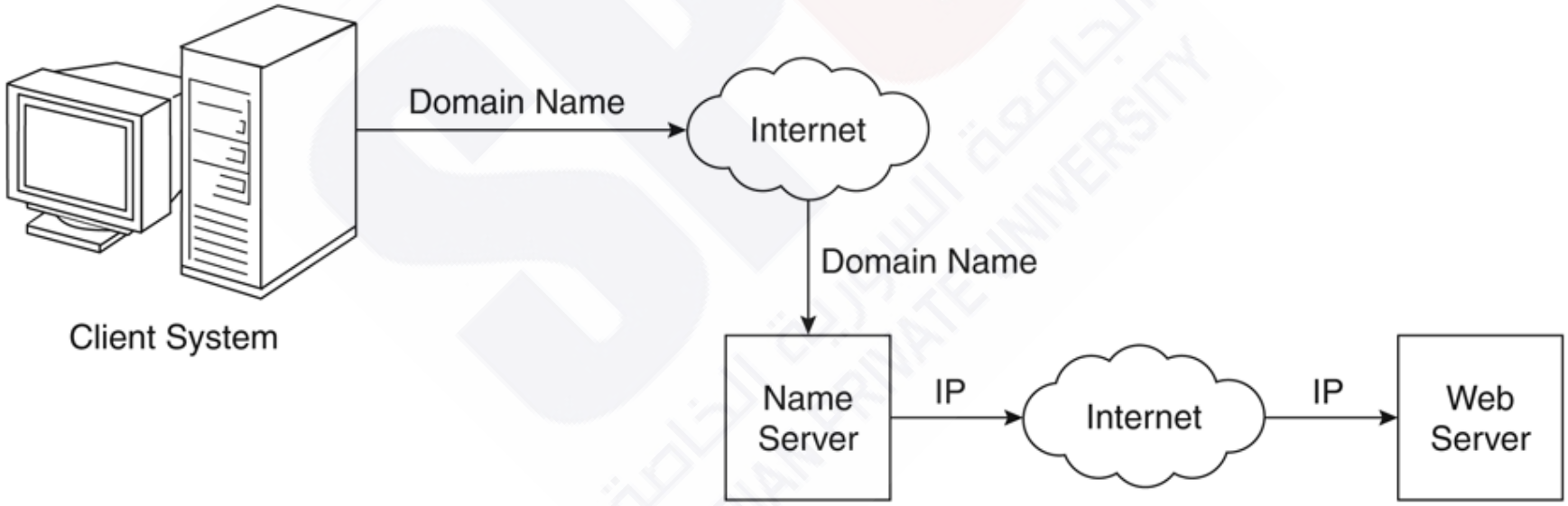
# Internet Protocols

- Internet Protocol (IP) Addresses
  - Every node has a unique numeric address
  - Form: 32-bit binary number
    - New standard, IPv6, has 128 bits (1998)
- Organizations are assigned groups of IPs for their computers
- Problem: By the mid-1980s, several different protocols had been invented and were being used on the Internet, all with different user interfaces (Telnet, FTP, Usenet, mailto

# Internet Protocols

- Domain names
  - Form: host-name.domain-names
  - First domain is the smallest; last is the largest
  - Last domain specifies the type of organization
  - Fully qualified domain name - the host name and all of the domain names
  - DNS servers - convert fully qualified domain names to IPs

# Domain name conversion



# Client and Server

- Clients and Servers are programs that communicate with each other over the Internet
- A Server runs continuously, waiting to be contacted by a Client
  - Each Server provides certain services
  - Services include providing web pages
- A Client will send a message to a Server requesting the service provided by that server
  - The client will usually provide some information, parameters, with the request

# The World-Wide Web

- A possible solution to the proliferation of different protocols being used on the Internet
- OriginTim Berners-Lee at CERN proposed the Web in 1989
- **Purpose**: to allow scientists to have access to many databases of scientific work through their own computers:
  - Document form: hypertext
  - Pages? Documents? Resources?
    - We'll call them documents
  - Hypermedia – more than just text – images, sound, etc.

# The World-Wide Web

- **Web or Internet?**
  - The Web uses one of the protocols, http, that runs on the Internet--there are several others (telnet, mailto, etc.)

# Web Browsers

- Browsers are clients - always initiate, servers react (although sometimes servers require responses)
- Mosaic - NCSA (Univ. of Illinois), in early 1993
  - First to use a GUI, led to explosion of Web use
  - Initially for X-Windows, under UNIX, but was ported to other platforms by late 1993
- Most requests are for existing documents, using HyperText Transfer Protocol (HTTP)
  - But some requests are for program execution, with the output being returned as a document

# Web Servers

- Provide responses to browser requests, either existing documents or dynamically built documents
- Browser-server connection is now maintained through more than one request-response cycle
- All communications between browsers and servers use Hypertext Transfer Protocol (HTTP)

# Web Server Operation

- Web servers run as background processes in the operating system
  - Monitor a communications port on the host, accepting HTTP messages when they appear
- All current Web servers came from either
  - The original from CERN
  - The second one, from NCSA

# Web Server Operation Details

- Web servers have two main directories:
  - Document root (servable documents)
  - Server root (server system software)
- Document root is accessed indirectly by clients
  - Its actual location is set by the server configuration file
  - Requests are mapped to the actual location
- Virtual document trees
- Virtual hosts
- Proxy servers
- Web servers now support other Internet protocols

# Web Server Operation : Apache

- Apache (open source, fast, reliable)
  - Directives (operation control):
    - ServerName
    - ServerRoot
    - ServerAdmin,
    - DocumentRoot
    - Alias
    - Redirect
    - DirectoryIndex
    - UserDir

# Web Server Operation : IIS

- IIS
  - Operation is maintained through a program with a GUI interface

# URLs

- General form:

  scheme:object-address

  – The scheme is often a communications protocol, such as http, telnet or ftp

- For the http protocol, the object-address is: fully qualified domain name/doc path
- For the file protocol, only the doc path is needed

# URLs

- Host name may include a port number, as in zeppo:80 (80 is the default, so this is silly)
- URLs cannot include spaces or any of a collection of other special characters (semicolons, colons, ...)
- The doc path may be abbreviated as a partial path
  - The rest is furnished by the server configuration
- If the doc path ends with a slash, it means it is a directory

# Multipurpose Internet Mail Extensions (MIME)

- Originally developed for email
- Used to specify to the browser the form of a file returned by the server (attached by the server to the beginning of the document)
- Type specifications
  - Form:

    type/subtype
  - Examples: text/plain, text/html, image/gif, image/jpeg

# Multipurpose Internet Mail Extensions (MIME)

- Server gets type from the requested file name's suffix (.html implies text/html)
- Browser gets the type explicitly from the server
- Experimental types
- Subtype begins with x-
- e.g., video/x-msvideo
- Experimental types require the server to send a helper application or plug-in so the browser can deal with the file

# The HyperText Transfer Protocol

- The protocol used by ALL Web communications
- **Request Phase**
  - The General Form of HTTP Request:

    HTTP method domain part of URL HTTP ver.

    Header fields

    blank line

    Message body
  - An example of the first line of a request:

    GET /degrees.html HTTP/1.1

# The HyperText Transfer Protocol: Methods

- **GET** - Fetch a document
- **POST** - Execute the document, using the data in body
- **HEAD** - Fetch just the header of the document
- **PUT -** Store a new document on the server
- **DELETE** - Remove a document from the server

# HTTP Response

- The General Form of HTTP Response:

  Status line

  Response header fields

  blank line

  Response body
- Status line format:

  HTTP version  status code  explanation
- Example: HTTP/1.1  200  OK

# The Web Programmer's Toolbox

- Document languages and programming languages that are the building blocks of the web and web programming
  - **XHTML**
  - Plug-ins
  - Filters
  - XML
  - **Javascript**
  - Java, Perl, Ruby, **PHP**

# XHTML

- To describe the general form and layout of documents
- An XHTML document is a mix of content and controls
  - **Controls** are tags and their attributes.
  - **Tags** often delimit content and specify something about how the content should be arranged in the document
  - **Attributes** provide additional information about the content of a tag

# Creating XHTML documents

- XHTML editors - make document creation easier
  - Shortcuts to typing tag names, spell-checker,
- WYSIWYG XHTML editors
  - Need not know XHTML to create XHTML documents

# Plugins and Filters

- **Plug ins**
  - Integrated into tools like word processors, effectively converting them to WYSIWYG XHTML editors
- **Filters**
  - Convert documents in other formats to XHTML

# Plugins and Filters: Advantages and Disadvantages

- **Advantages** of both filters and plug-ins:
  - Existing documents produced with other tools can be converted to XHTML documents
  - Use a tool you already know to produce XHTML
- **Disadvantages** of both filters and plug-ins:
  - XHTML output of both is not perfect - must be fine tuned
  - XHTML may be non-standard
  - You have two versions of the document, which are difficult to synchronize

# XML

- A meta-markup language
- Used to create a new markup language for a particular purpose or area
- Because the tags are designed for a specific area, they can be meaningful
- No presentation details
- A simple and universal way of representing data of any textual kind

# JavaScript

- A client-side HTML-embedded scripting language
- Only related to Java through syntax
- Dynamically typed and not object-oriented
- Provides a way to access elements of HTML documents and dynamically change them

# Java

- General purpose object-oriented programming language
- Based on C++, but simpler and safer
- Our focus is on applets, servlets, and JSP

# Perl

- Provides server-side computation for HTML through CGI
- Perl is good for CGI programming because:
  - Direct access to operating systems functions
  - Powerful character string pattern-matching operations
  - Access to database systems
- Perl is highly platform independent, and has been ported to all common platforms
- Perl is not just for CGI

# PHP

- A server-side scripting language
- An alternative to CGI
- Similar to JavaScript
- Great for form processing and database access through the Web