

1

CSC 443: Web Programming

Haidar Harmanani

Department of Computer Science and Mathematics
Lebanese American University
Byblos, 1401 2010 Lebanon

CSC443: Web Programming

2

The INTERNET... and a bit of history

Haidar Harmanani

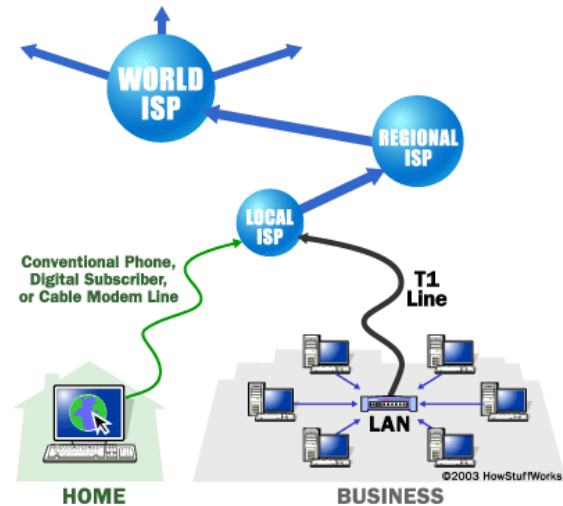
Department of Computer Science and Mathematics
Lebanese American University
Byblos, 1401 2010 Lebanon

CSC443: Web Programming

The Internet and the Web

3

- A collection of computer networks that use a protocol to exchange data
- **Are they the same thing?**

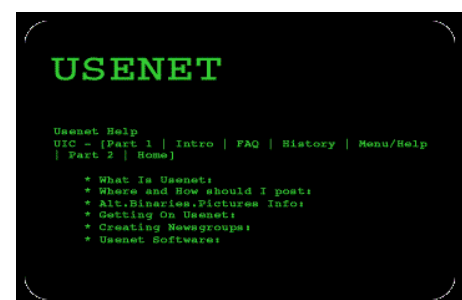
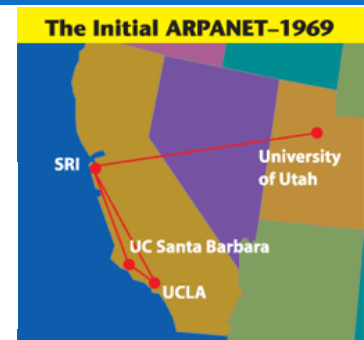


CSC443: Web Programming

Brief history

4

- Began as a US Department of Defense network called ARPANET (1960s)
- Packet switching (in the 60s)
- E-mail is born on 1971
- TCP/IP beginning on 1974 (Vinton Cerf)
- USENET (1979)
- By 1987: Internet includes nearly 30,000 hosts



CSC443: Web Programming

History of Internet [8 Minutes]

5



CSC443: Web Programming

Brief history (cont.)

6

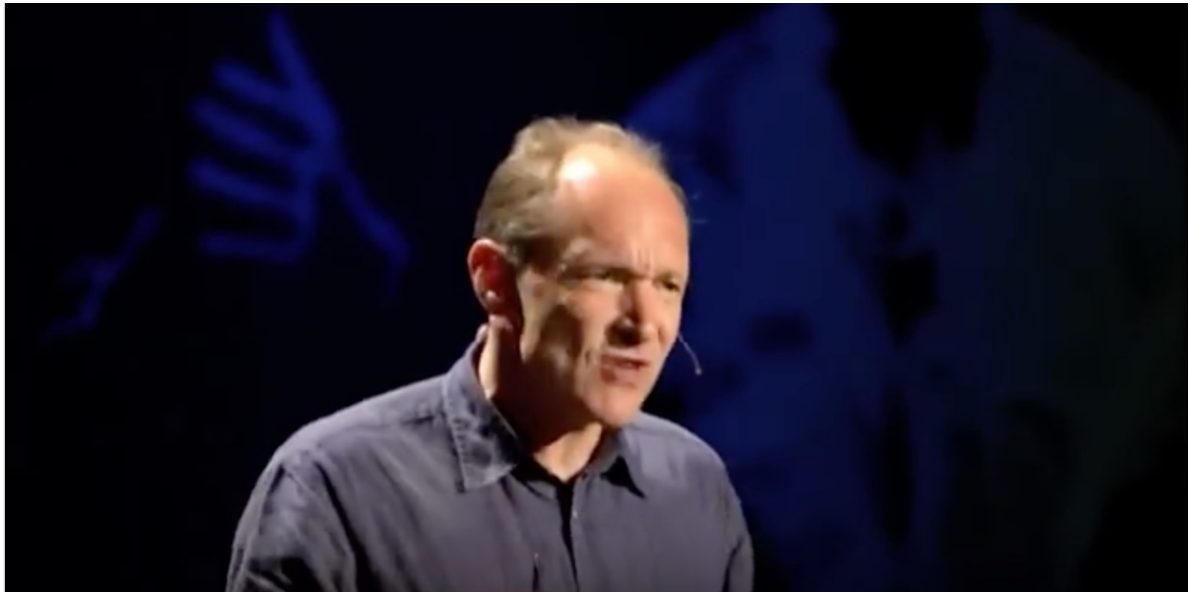
- ❑ WWW created in 1989-91 by Tim Berners-Lee
- ❑ Popular web browsers released:
 - ▣ Netscape 1994
 - ▣ IE 1995
- ❑ Amazon.com opens in 1995
- ❑ Google January 1996
- ❑ Wikipedia launched in 2001
- ❑ MySpace opens in 2003
- ❑ Facebook February 2004



CSC443: Web Programming

Brief history (cont.)

7



CSC443: Web Programming



8

CSC443: Web Programming

The future of the internet?

9



CSC443: Web Programming

Key aspects of the internet

10

- ☐ Sub-networks are independent
- ☐ Computers can dynamically join and leave the network
- ☐ Built on open standards
- ☐ Lack of centralized control (mostly)
- ☐ Everyone can use it with simple, commonly available software

CSC443: Web Programming

People and organizations

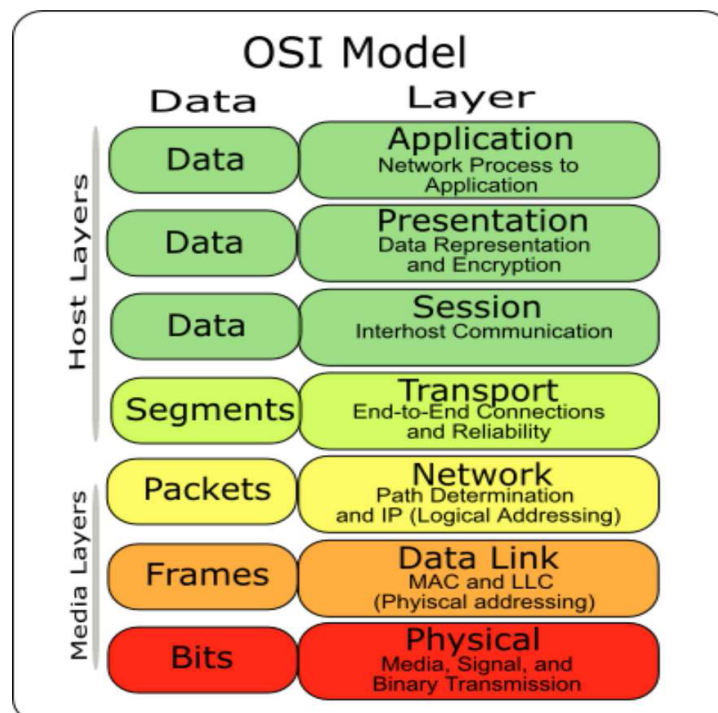
11

- Internet Engineering Task Force (IETF): internet protocol standards
- Internet Corporation for Assigned Names and Numbers (ICANN): decides top-level domain names
- World Wide Web Consortium (W3C): web standards



CSC443: Web Programming

Layered architecture



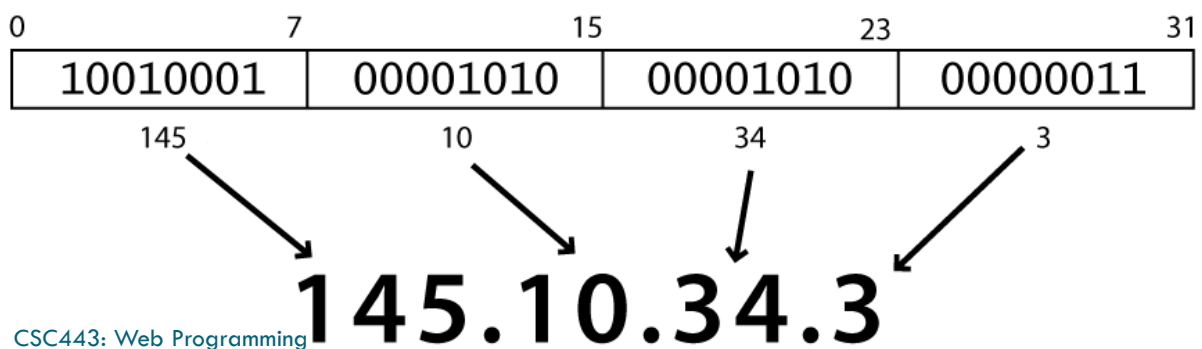
12

CSC443: Web Programming

Internet Protocol (IP)

13

- Simple protocol for data exchange between computers
- IP Addresses:
 - ▣ 32-bit for IPv5
 - ▣ 128-bit for IPv6



Transmission Control Protocol (TCP)

14

- Adds multiplexing, guaranteed message delivery on top of IP
- Multiplexing: multiple programs using the same IP address
- Port: a number given to each program or service
 - ▣ port 80: web browser (port 443 for secure browsing)
 - ▣ port 25: email
 - ▣ port 22: ssh
- Some programs (games, streaming media programs) use simpler UDP protocol instead of TCP

Web Servers

15

- Web server: software that listens for web page requests
 - ▣ Apache
 - ▣ Microsoft Internet Information Server (IIS)



CSC443: Web Programming

Application Server

16

- Software framework that provides an environment where applications can run
 - ▣ Apache
 - ▣ Glassfish
 - ▣ WebSphere
 - ▣ WebLogic



CSC443: Web Programming

Inside a Google Data Center [5 Minutes]

17



CSC443: Web Programming

Web Browser

18

- Web browser: fetches/displays documents from web servers
 - ▣ Mozilla Firefox
 - ▣ Microsoft Internet Explorer (IE)
 - ▣ Apple Safari
 - ▣ Google Chrome
 - ▣ Opera

CSC443: Web Programming

Domain Name Server (DNS)

19

- Set of servers that map written names to IP addresses
 - ▣ Example: ju.edu → **204.29.160.73**
- Many systems maintain a local cache called a hosts file
 - ▣ Windows: C:\Windows\system32\drivers\etc\hosts
 - ▣ Mac: /private/etc/hosts
 - ▣ Linux: /etc/hosts
- <https://youtu.be/2ZUxoi7YNgs>

CSC443: Web Programming

Uniform Resource Locator (URL)

20

- Identifier for the location of a document on a web site
 - ▣ Example: http://dept.ju.edu/cs/index.html
- Upon entering this URL into the browser, it would:
 - ▣ ask the DNS server for the IP address of dept.ju.edu
 - ▣ connect to that IP address at port 80
 - ▣ ask the server to GET /cs/index.html
 - ▣ display the resulting page on the screen

CSC443: Web Programming

Hypertext Transport Protocol (HTTP)

21

- Set of commands understood by a web server and sent from a browser
- Some HTTP commands (your browser sends these internally):
 - ▣ GET filename : download
 - ▣ POST filename : send a web form response
 - ▣ PUT filename : upload
- Exercise: simulate a browser with a terminal window

CSC443: Web Programming

A Small Lab

22

- <http://vlsi.byblos.lau.edu.lb/classes/csc443/hello.html>
- <http://vlsi.byblos.lau.edu.lb/classes/csc443/image.jpg>
- yoda:~ haidar\$ nslookup
- > vlsi.byblos.lau.edu.lb
- Server: 192.168.1.1
- Address: 192.168.1.1#53

- Non-authoritative answer:
- Name: vlsi.byblos.lau.edu.lb
- Address: 80.77.185.6
- >

CSC443: Web Programming

A Small Lab

23

- ☐ telnet 80.77.185.6 80
- ☐ GET /classes/csc443/hello.html HTTP/1.0
- ☐ GET /classes/csc443/hello.html HTTP/1.1

- ☐ HEAD /classes/csc443/image.jpg HTTP/1.0
- ☐ GET /classes/csc443/image.jpg HTTP/1.1

- ☐ What do you see?
- ☐ Conclusion?

CSC443: Web Programming

A Small Lab

24

- ☐ HTTP/1.1 requires header definition

CSC443: Web Programming

The Internet of Things [Assg.]

25

