CSC 443: WEB PROGRAMMING

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

Today

- Course information
- Course Objectives
- A Tiny assignment
- Overview of the web area

Who Am I?

- Haidar M. Harmanani
  - Professor of Computer Science, Byblos
  - haidar@lau.edu.lb
  - 09.547254 x2360
  - 810 Block A
- Course Webpage:
  - http://vlsi.byblos.lau.edu.lb/csc443.html or
  - http://harmanani.github.io/csc443.html (mirror)

Course Overview

- Project-based course with a lot of self learning
- You have to eventually develop a web project
  - You choose the problem
  - You choose the technology
  - You choose the solution
  - You do what you want!
- Work in groups of 2
- Seminar and demo at the end
- Written report

Course Objectives

5

- At the end of this class you will be able to:
  - Design and implement a professional website
  - Author web pages using HTML
  - Make stylistic decisions with CSS
  - Create interactive websites with JavaScript and jQuery
  - Enhance interactive websites with AJAX, JSON, and XML
  - Use PHP for server programming
  - Write simple constructs in NodeJS, Angular, and Python

Course Objectives (cont.)

6

- At the end of this class you will be able to:
  - Understand the client-server programming model and apply this to your designs
  - Create your own web programming portfolio
  - Speak the web programming lingo
  - Have fun with web programming!

Grading

7

- **Midterm:** **25%**
- **Final:** **35%**
- **Individual Labs:** **10%**
- **Project:** **30%**
  - *Oral Presentation* *5%*
  - *Technical Merit* *15%*
  - *Technical Innovation* *5%*
  - *Written Report* *5%*

Web technologies we will talk about

8

- HTML 5
- CSS
- CGI – interface for running web apps
- Document Object Model
- JSON - JavaScript Object Notation
- MongoDB
- JBOSS – Application platform
- BEA – Application platform
- Jakarta – Application platform (Apache)
- Node.js and Express
- AJAX and JQuery

Textbooks Info

Content Management Systems

- A portal provides content authoring systems that allow non-technical staff to create content.
- Control access to content to allow only authorized users access to document repositories.
- In principle, Content Management Systems is a service (external application) which could be integrated within the portal
  - Wordpress
  - Drupal
  - Joomla

Lab 1

- Write an html page that gives some info about yourself by Thursday and host it on *github* pages!
  - See https://pages.github.com

Why did you take CSC 443?

Why do you need CSC443?

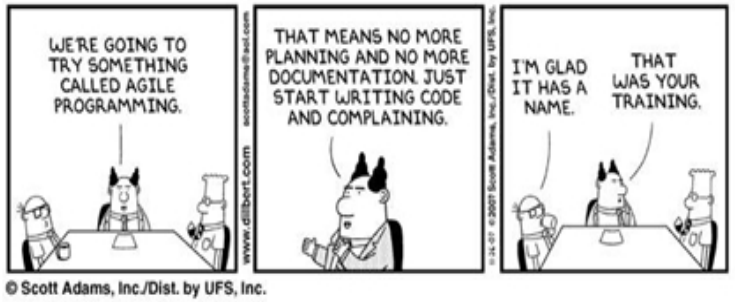
Lectures

- Check the schedule in the class webpage
- Read the syllabus and policies carefully
- Reading, assgs, and labs will be posted online
- We will be using piazza for questions an discussions
  - https://piazza.com/lau.edu.lb/fall2016/csc443/home

Lectures

- Lectures will be interactive. This means:
  - You will need to study the new material before every lecture (slides, book, and online material)
  - We will have a lab on every lecture, so you will need to code in almost every lecture
  - You will post your questions on the discussion board (on piazza) before each lecture. If you do not post any questions, I assume you have understood everything. Therefore…
  - You may be called in class to explain the material to your classmates

Programming Project



Programming Project

- You can start working on this from the first week of the class
- Design and implementation of a professional website:
  - Professional Style
  - Interactive
- You can complete the project in teams of two

Programming Project

- Your project should include at least four of the following grouped items with the first item being compulsory:
  - HTML, CSS, secure forms, JavaScript, AJAX, JSON, cookies, and sessions;
  - A Client-Side Framework such as jQuery or Angular;
  - Node.JS;
  - A non-structured database such as MongoDB or NoSQL;
  - PHP and MySQL;
  - A Web Service - use an external web service, mashed up with your own application to create something even better.
- Your project should document the use of these technologies!

Programming Project

Design & Evaluate - think carefully about how users will use your site, design a great interface, and evaluate it with real people.

Go Mobile - create a version of your project designed to go mobile.

Server-Side Processing - do processing on the server to prepare for user requests in advance.

Homework

- It will involve:
  - Applying what we learned in class
  - Clean design and coding
  - Clear documentation
- Homework will be completed *individually*

Policies

21

- Cheating means "submitting, without proper attribution, any computer code that is directly traceable to the computer code written by another person."
- Or even better:
  - "Any form of cheating, including concealed notes during exams, copying or allowing others to copy from an exam, students substituting for one another in exams, submission of another person's work for evaluation, preparing work for another person's submission, unauthorized collaboration on an assignment, submission of the same or substantially similar work for two courses without the permission of the professors. Plagiarism is a form of Academic Misconduct that involves taking either direct quotes or slightly altered, paraphrased material from a source without proper citations and thereby failing to credit the original author. Cutting and pasting from any source including the Internet, as well as purchasing papers, are forms of plagiarism."
- I give students a failing homework grade for any cheating.
- A second cheating attempt will be in an F in the course!

Policies

22

- You may discuss homework problems with classmates, after you have made a serious effort in trying the homework on your own.
- You can use ideas from the literature (with proper citation).
- You can use anything from the textbooks/notes.
- The code you submit **must be written completely by you.**

Policies

23

- Late policy:
  - 5% is reduced by every day the homework is late

24

The INTERNET… and a bit of history

What is the internet?

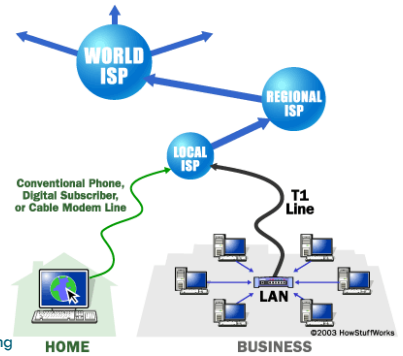
25

- A “series of tubes”
- How many Internets are out there?
- Is Google one of them?

What is the internet?

26

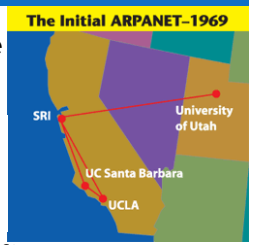
- A collection of computer networks that use a protocol to exchange data
- Is the World Wide Web (WWW) and the internet the same?



Brief history

27

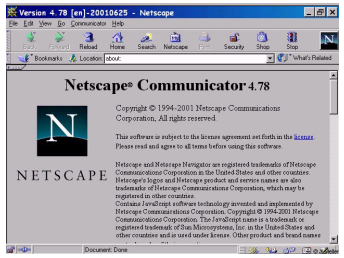
- Began as a US Department of Defense network called ARPANET (1960s-70s)
- Packet switching (in the 60s)
- E-mail is born on 1971
- TCP/IP beginning on 1974 (Vinton Cerf)
- USENET (1979)
- By 1987: Internet includes nearly 30,000 hosts



Brief history (cont.)

28

- 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



The future of the internet?

Key aspects of the internet

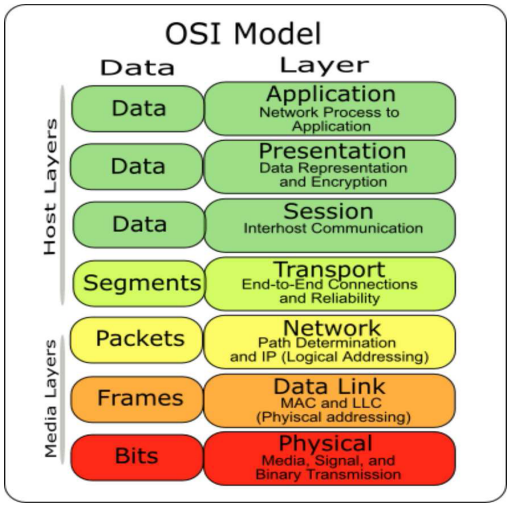
- Sub-networks are independent
- Computers can dynamically join and leave the network
- Built on open standards
- Lack of centralized control (mostly)
- Everyone can use it with simple, commonly available software

People and organizations

- 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

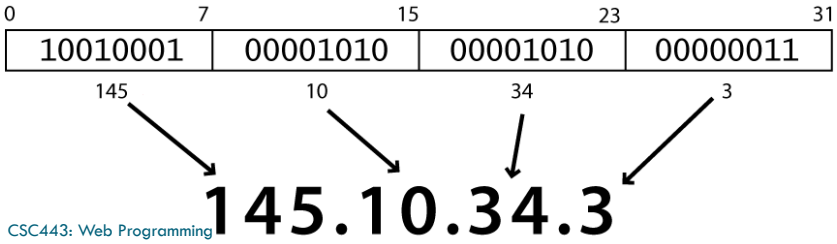
IETF ICANN W3C

Layered architecture



Internet Protocol (IP)

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

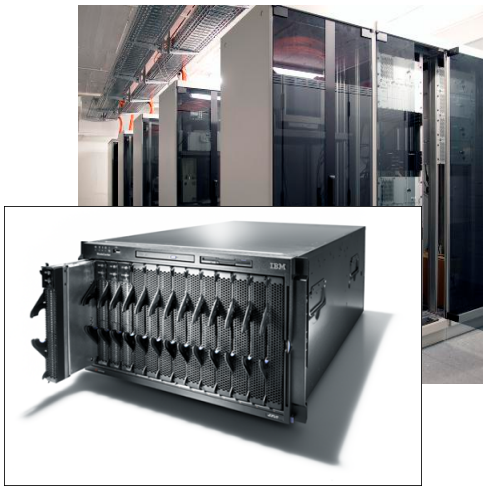


Transmission Control Protocol (TCP)

- 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

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



Application Server

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

Web Browser

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

Domain Name Server (DNS)

- 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

Uniform Resource Locator (URL)

- 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

Hypertext Transport Protocol (HTTP)

- 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

HTTP Error Codes

- When something goes wrong, the web server returns a special "error code" number
- Common error codes:

Number	Meaning
200	OK
301-303	page has moved (permanently or temporarily)
403	you are forbidden to access this page
404	page not found
500	internal server error

Internet Media ("MIME") types

MIME type	file extension
text/html	.html
text/plain	.txt
image/gif	.gif
image/jpeg	.jpg
video/quicktime	.mov
application/octet-stream	.exe

Web Languages

- Hypertext Markup Language (HTML): used for writing web pages
- Cascading Style Sheets (CSS): stylistic info for web pages
- PHP Hypertext Processor (PHP): dynamically create pages on a web server
- JavaScript: interactive and programmable web pages

Web Languages(cont.)

- Asynchronous JavaScript and XML (Ajax): accessing data for web applications
- eXtensible Markup Language (XML): metalanguage for organizing data