

CSC435: Web Programming

Lecture 1: the internet and the web

Bei Xiao

American University

Jan 15, 2019

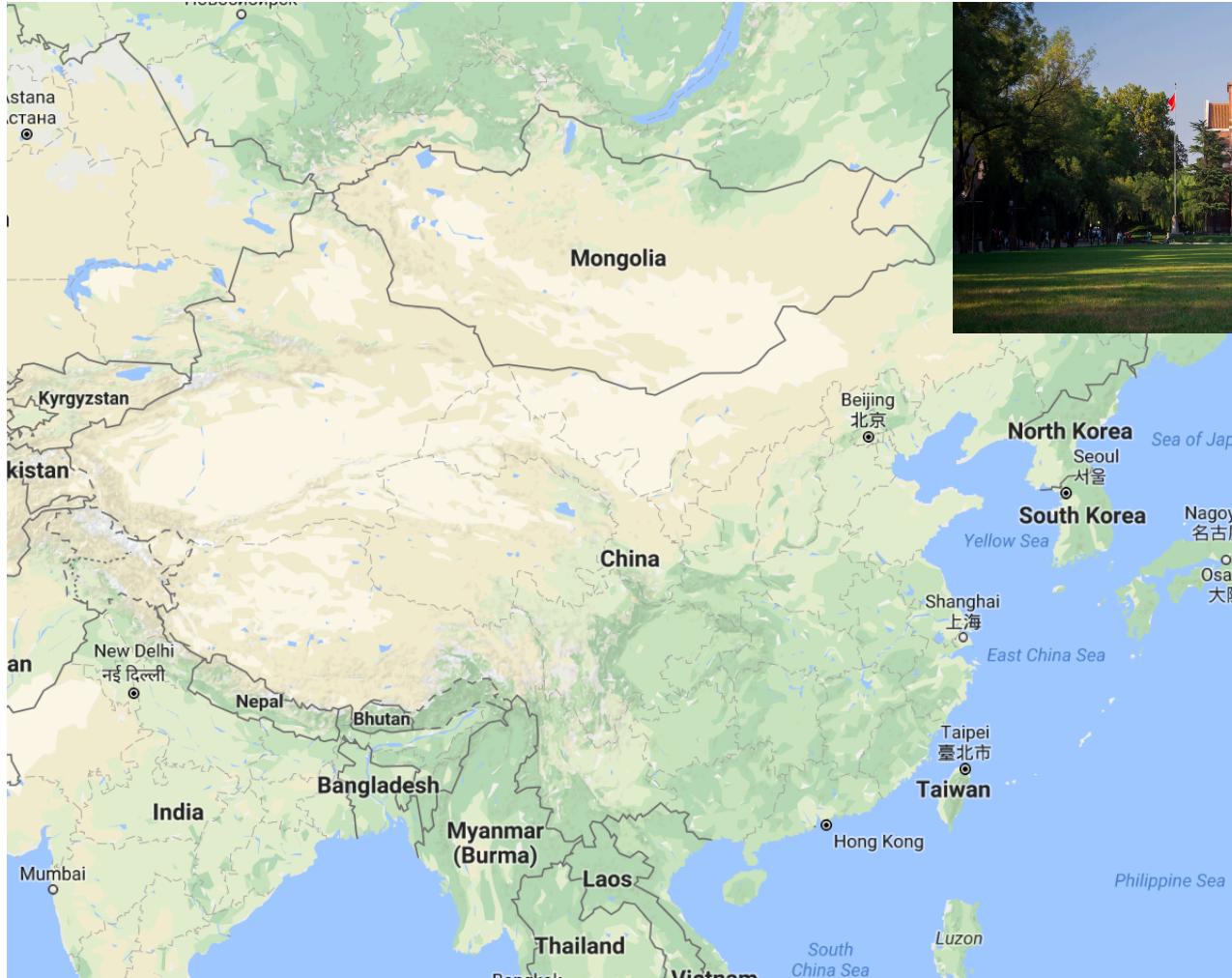
Course Info

- Instructor: Prof. Xiao, bxiao@american.edu
- Office: Don Myers Room 110
- Office hour: Tuesday, 3-4pm, Friday, 3-4pm
- Class time: Jan 15- April 30th
- Prerequisite: Intro to Programming (CSC 280)
- Course website:
- **<https://github.com/fruittree/CSC435WebProgramming>**

About Your Professor

- **Prof. Bei Xiao (Professor of Computer Science):**
B.S. Tsinghua University, PhD University of Pennsylvania, Postdoc MIT.
<https://sites.google.com/site/beixiao/>
- Research area: Human and Computer Vision. Computer Graphics.
- Specialty: Material Perception, Multisensory perception, Physical-realistic rendering.
- Preferred language: Python, MATLAB, JavaScript.
- Intro to programming (CSC 280 last semester),
- Computer Vision (CSC476, Every year),
- Web Programming (This semester)
- Computer graphics (near future)

A bit about me



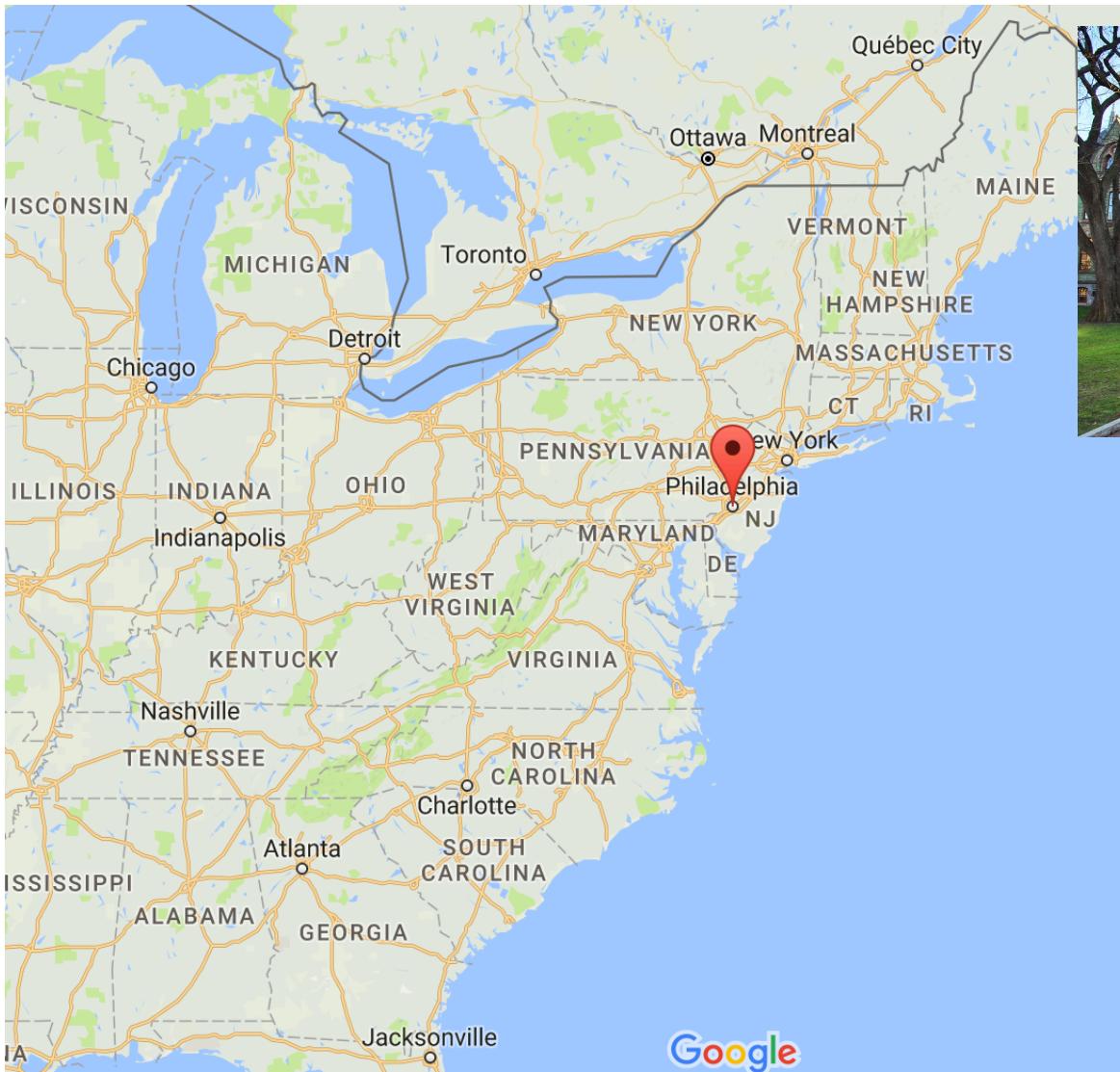
College:

Tsinghua
University

Beijing, China

Degree: Chemistry

A bit about me



Graduate School:

University of
Pennsylvania

Philadelphia, PA

Degree: Computational
neuroscience

A bit about me



Postdoctoral
Fellowship:

MIT

Cambridge, MA

Computer Vision and
Human Perception

A bit about me



Summer Research 2018

University of
Tuebingen

Tuebingen, Germany

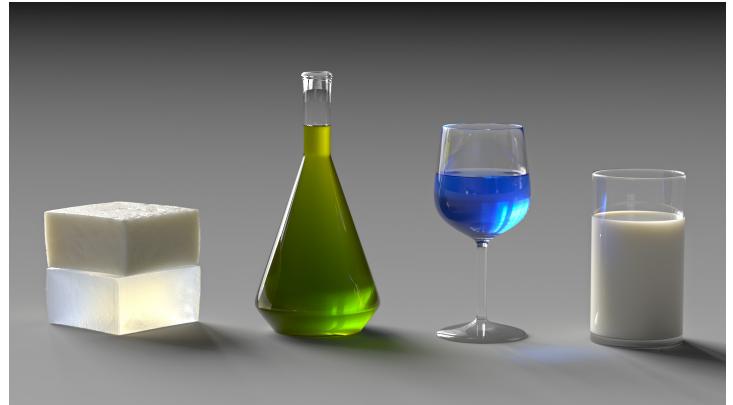
Human Perception
Deep Learning
Machine Vision

My research: vision and graphics

Machine Inference of material properties of cloth



Rendering and perception of translucent materials



Visual and tactile material categorization



Tactile perception in VR



This class is about

- Foundation of The Internet.
- How the Internet works
- How to make web pages that is interactive.
- How web page works.

This class is NOT about

- How to make website pretty and stylish
- How to make mobile apps
- How to make website user friendly
- How to display and visualize data
- If you wish to learn these things, take courses in graphic design program and DataVis calss from Alex Godwin

Promises

- Web programming is in real-time is hard and messy.
- I won't know the answers to lots of questions, and I will google things in front of you.
- I will fix my terrible code in front of you.

Grading

- 60% individual assignments (approximately 5-6 projects)
- 10% Mid-term exam (open-book programming exercises), Time: the Friday after Spring break.
- 10% Creative project
- 10% Final Project
- 10% In-class exercises and quiz (every class, both results and participation count!).
- All projects are independent unless teamwork is specified.

**Key word:
Programming**

Creative Project

- Unlike a usual unfinished crammed final project that assigned at the end of the semester..... This term starts with **building a web app from scratch on the first day!**
- Example: a dynamic website that collects user responses from the crowds (e.g. rating DC neighborhood safety, Tracking election data.)
- Entirely your own design and topic.
- Every week you finish some specified requirement (you spend one hour each week).
- But have to turn in whatever you have.
- And the end, you have a working web app that you can upload to Amazon Mechanical Turk and present your results.

Late Policy

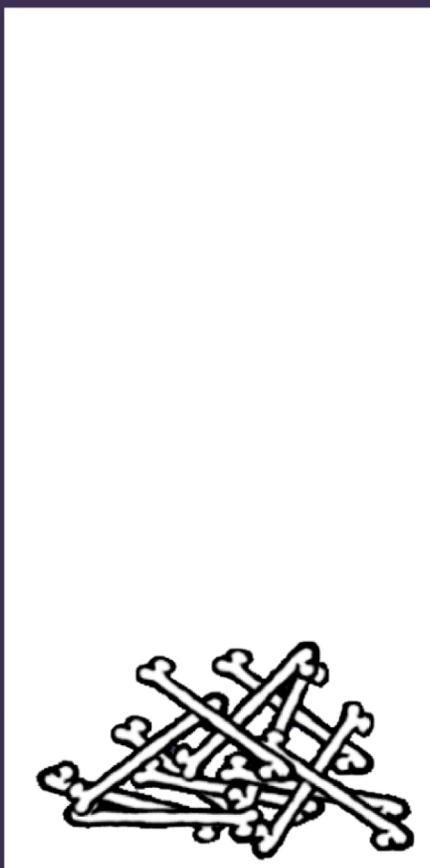
- Read Course Syllabus!
- Every late homework will receive 5% automatic reduction of grade for every late day.
- No homework is accepted **5 days after** the due date and after the final exam date (will receive zero).

Overview

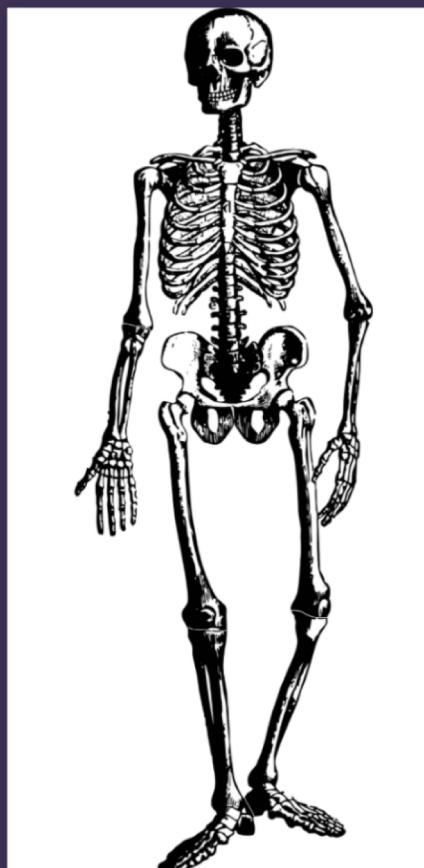
- Composition of a website
- What is the internet?
- How to access a website?
- What is an IP address
- What is URL?
- How to send request to server?
- What are client side technologies?
- What are server side technologies?

What is a web page, really?

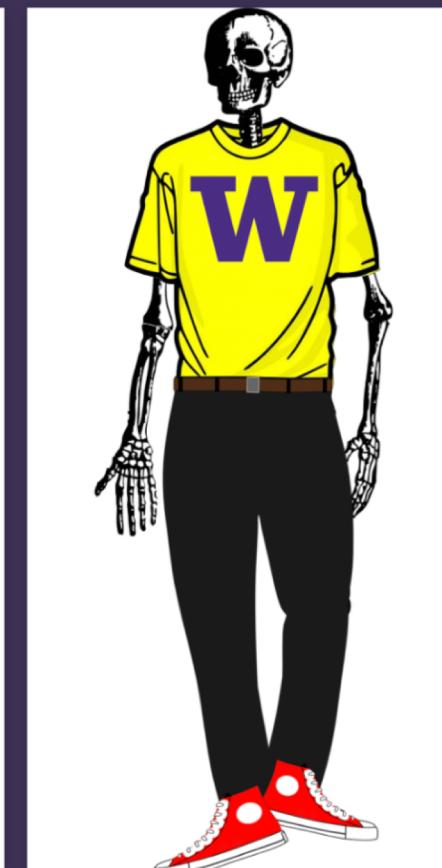
Content



Structure



Style



Behavior



Words and images

HTML

CSS

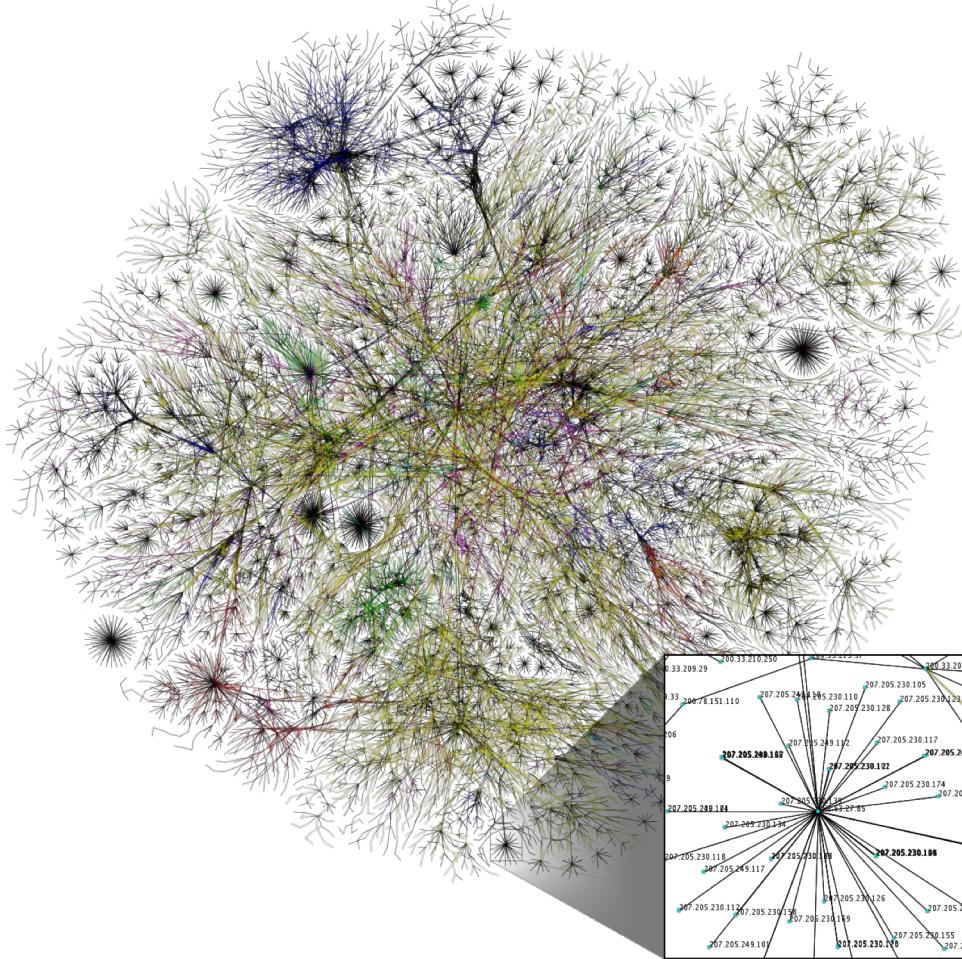
Javascript & Server

Web Languages/Techologies

Big list of different things

- Content: Hypertext Markup Language ([HTML](#))
- Style: Cascading Style Sheets ([CSS](#))
- Behavior: [JavaScript](#) and libraries for creating interactive web pages
- Data representation and transfer: JavaScript Object Notation([JSON](#)).
- Dynamic web servers: Hypertext Process ([PHP](#))
- Enhanced data transfer: Asynchronous JavaScript and XML ([Ajax](#))
- Data Bases: Structured Query Language ([SQL](#))

What is the internet?



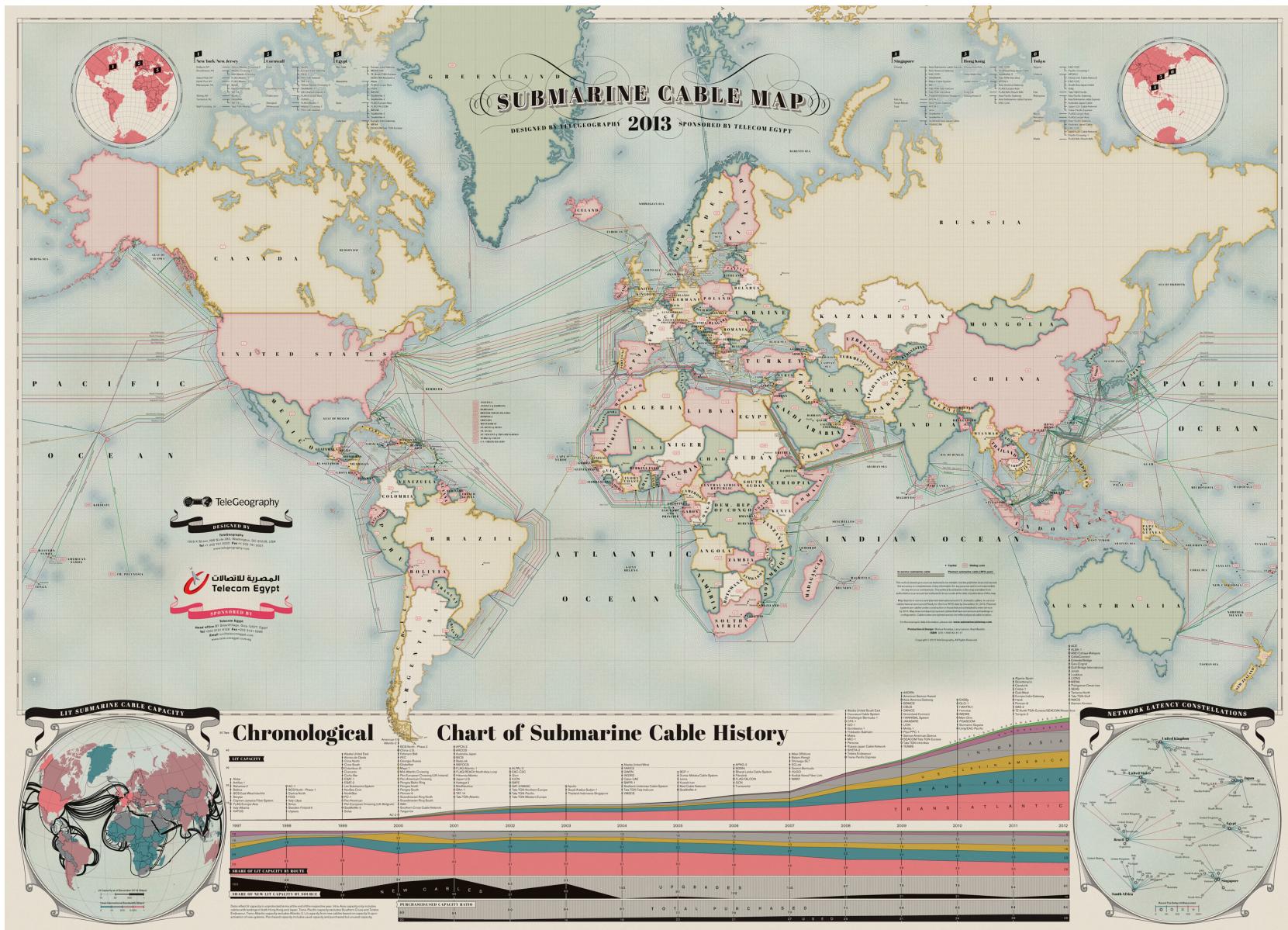
http://en.wikipedia.org/wiki/Opte_Project

What is the internet?

- The internet is a physical network connecting millions of computers using the same protocols (TCP/IP) of sharing transmitting information. In reality, internet is network of small networks.

The internet?

- Wikipedia: <https://en.wikipedia.org/wiki/Internet>
- A connection of computer networks the internet protocol (IP)
- Layers of communication protocols: IP-> TCP/UDP-> HTTP/POP/FTP
- What is the difference between Internet and the World Wide Web (WWW)?



<http://www.telegeography.com/telecom-resources/map-gallery/submarine-cable-map-2013/>

Brief history of the internet

- Began as US Department of Defense network called APPANET (1960s-70s).
- Initial services: electronic mail, file transfer through a computer
- Opened to commercial interests in late 80s.
- WWW created in 1989-91 by Tim Berners-Lee (2016 Turing Award)
- Popular web browsers released: Netscape 1994, IE 1995, Mozilla 1998, Safari, 2003, Firefox, 2004, Chrome 2011.
- Amazon.com opens in 1995; Google January 1999.
- Facebook opened in 2004, Twitter in 2006.
- Snapchat created in 2011.

Fun facts about the internet

- 1 out of 4 people have internet access.
- The internet reaches 50 million users in 5 years.
- More videos are uploaded to YouTube in the last 2 months than had ABC, NBC, and CBS all airing new content 24/7 since 1948.
- There are 121,013,927 unique domains.
- Most domains and pages are saved on servers and servers are saved on massive data centers.
- Worldwide average internet speed increased to 7.2Mbps. Average speed has increased from 5.6Mbps in Q4 of 2015 to now. South Korea (28.6) retains 1st place for internet speed in 2017. United States (18.7).

Key Aspects of the Internet

- Subnetworks can stand on their own
- Computers can dynamically join and leave the network.
- Built on open standards; anyone can create a new internet device.
- Lack of centralized control (mostly)
- Everyone can use it with simple, commonly available software.

Question

- Who “runs” the internet? Who is responsible for overseeing it?

People and organizations

- Internet Engineering Task Force ([IETF](#)):internet protocol standards
- Internet Corporation for Assigned Names and Numbers ([ICANN](#)): decides top-level [domain name](#).
- World Wide Web consortium ([W3C](#)): web standards.
- These protocols are carried out in large part by Internet service providers and other companies and organizations who build internet related projects and applications.

Layered Architecture

- Physical layer: devices such as ethernet, coaxial cables, fiber-optical lines, modems
- Data Link Layer: basic hardware protocols (ethernet, wifi, DSL PP).
- Network/Internet Layer: basic software protocol (IP)
- Transport layer: adds reliability to network layer (TCP, UDP)
- Application layer: implements specific communications for each kind of program (HTTP, POP3/IMAP, SSH, FTP)

The World Wide Web (WWW)

- World wide web: a collection of **interlinked multimedia documents** that are stored on the Internet and accessed using common HTTP and HTTPS protocols.
- E.g. <https://www.facebook.com/>
- How is it different from the Internet?

The World Wide Web

- **Web server**: software that listens for web page requests
 - Apache
 - Microsoft Internet Information Server (IIS)(part of windows)
 - NGINX
- Web Browsers: fetches/display documents from web server.
 - Mozilla Firefox
 - Microsoft IE and Microsoft Edge
 - Apple Safari
 - Google Chrome
 - Opera

What are other internet-based applications?

- Email
- Telnet
- FTP
- Instant messaging
- File sharing
- Data storage

Quiz

- Is the internet the same as the web?

Quiz

- Is the internet the same as the web?
- If not, what is the key distinction?

Quiz

- Is the internet the same as the web?
- If not, what is the key distinction
- Answer: Internet is hardware; web is software along with data.

Accessing a website

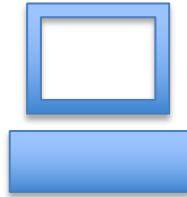
- Type in a URL in an address bar in your browser:
www.xkcd.com
- “/” refers path of the server.
- <https://myau.american.edu/my.policy>

AU60185:CSC435_WebProgramming bxiao\$ Ping
american.edu

PING american.edu (147.9.4.186): 56 data bytes
Request timeout for icmp_seq 0

Accessing a website

- Type in a URL in an address bar in your browser: www.xkcd.com
- Web server: a machine that stores all the content such as images and videos



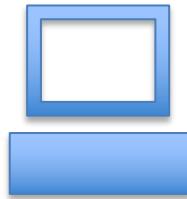
Client
Your
Browser



Web server

Accessing a website

- Computer communicate the server to tell which webpage you want to view
- Server process the data and return the web content the browser needs



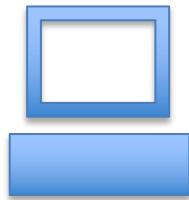
Client
Your
Browser



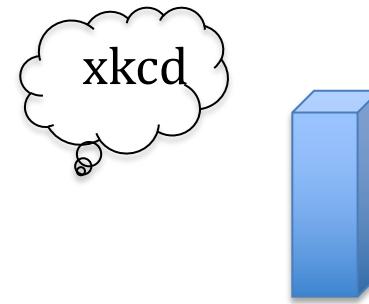
Web server

Accessing a website

- How does the browser know which server contains contents of the website xkcd?

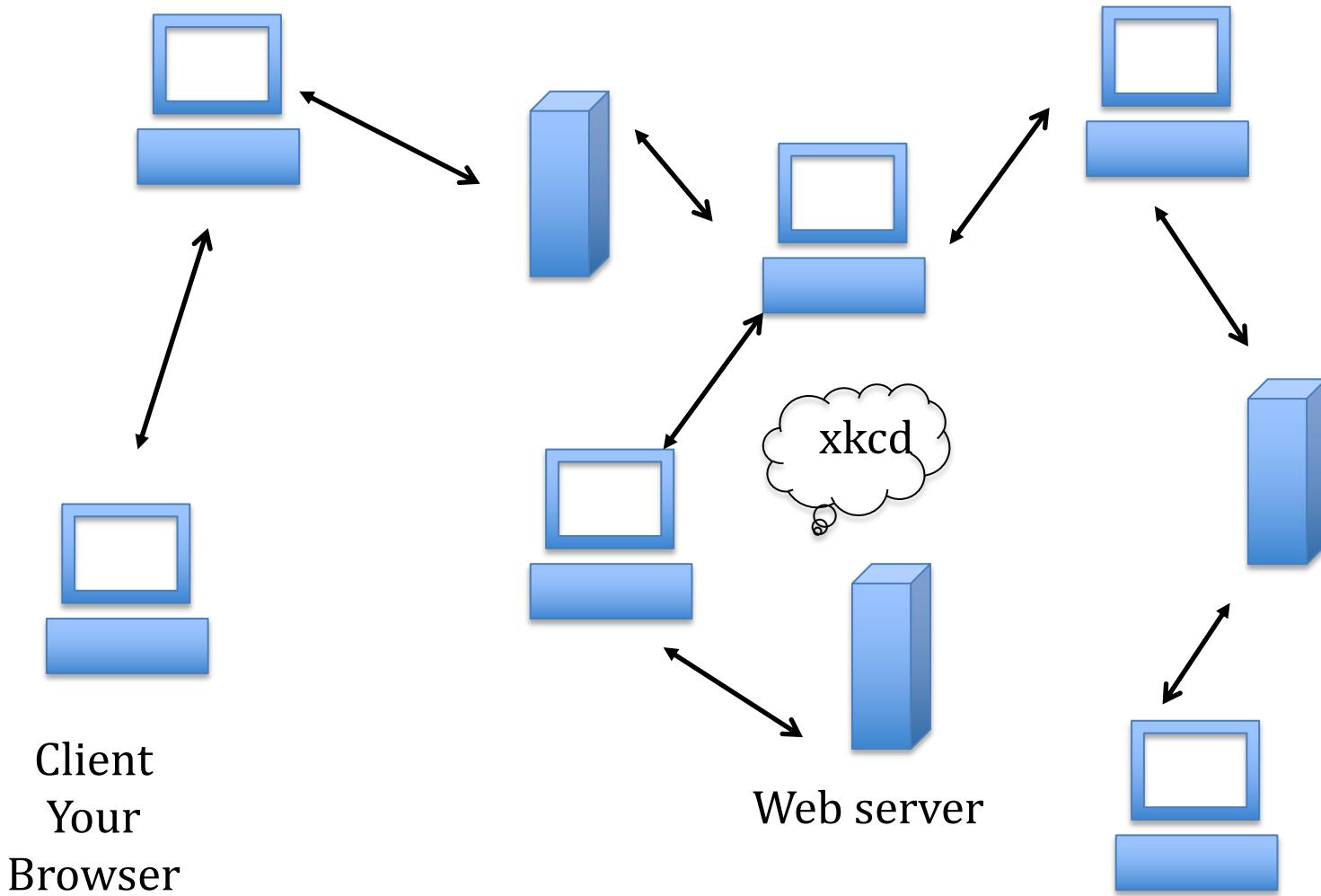


Client
Your
Browser

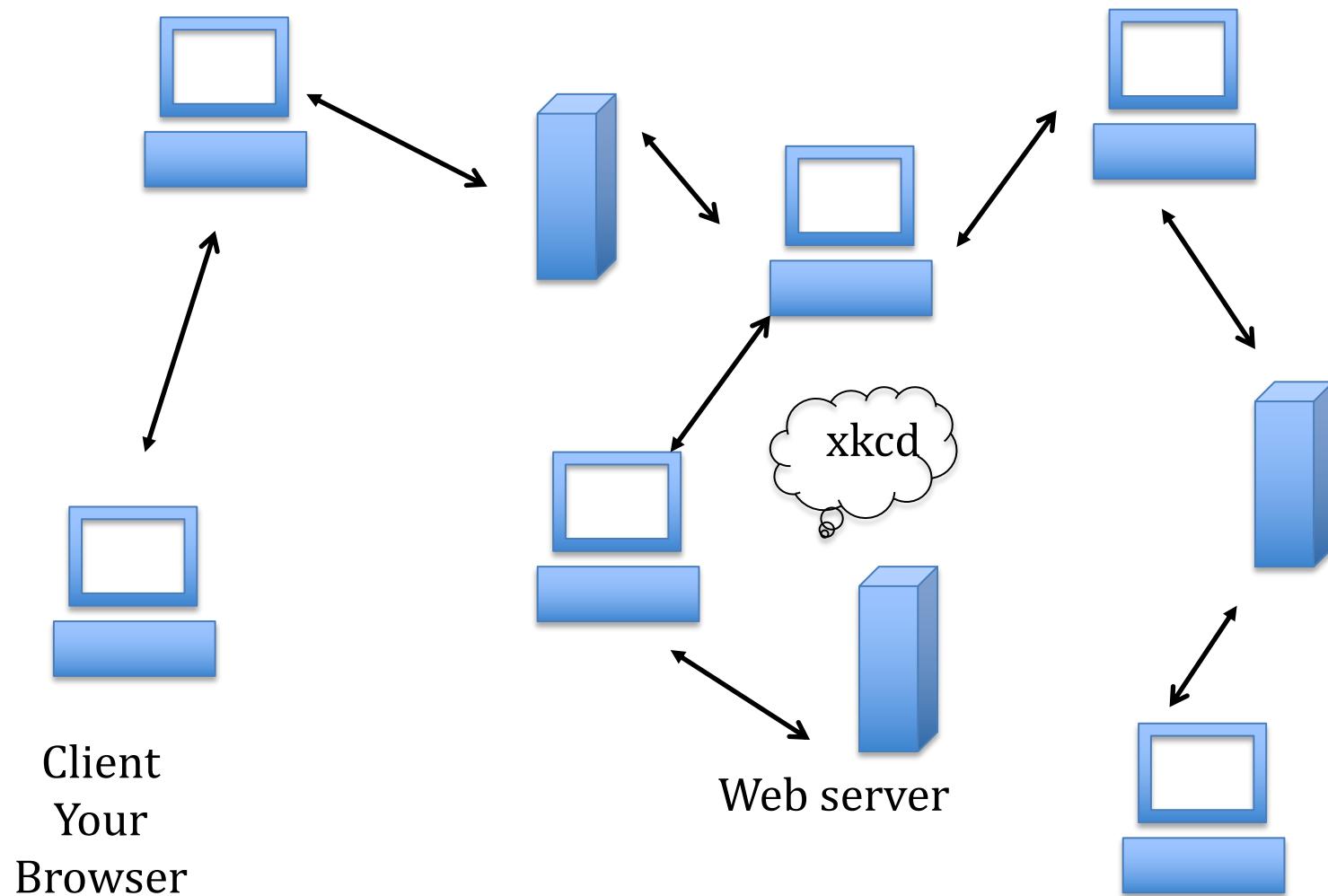


Web server

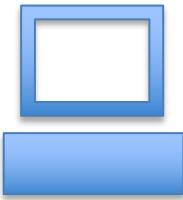
The Internet



Which server has the content of the requested website?



IP Address (which computer and where it is located)



Client
Your
Browser



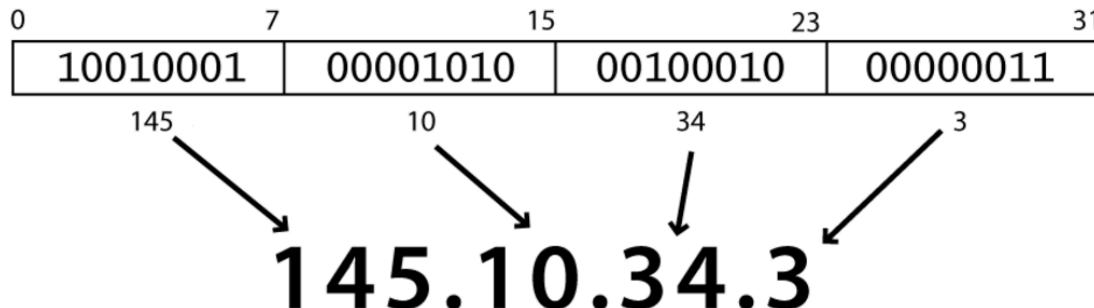
Web server
IP: 72.26.203.99

IP (Internet Protocol) Address

- The IP address give the location of the computer so that your computer can reach the server. Every computer at AU has the IP address from 147.
- It contains 4 numbers, ranging from 1 to 255.
- For more information, look at:[Internet Protocol at wikipedia](#)

IPv4 (Internet Protocol)

- A simple protocol for attempting to send data between two computers.
- Each device has a 32-bit IP address written for four 8-bit numbers (0-255)
- Find out your local IP address
 - In a terminal, type ipconfig(Windows) and ifconfig (Mac)
- Find out your internet IP address: whatismyip.com



IP address versus URL

- IP address: the actual address of a targeted web server.
- URL refers to Uniform Resource Locator
- URL: human-readable pseudonym for a web-server.
- URL is mapped into an IP by a DNS server.

Question

- Why don't we directly type IP address into the browser?

Answer

- URLs allow users to change their web server without remembering the IP address

- Further reading:

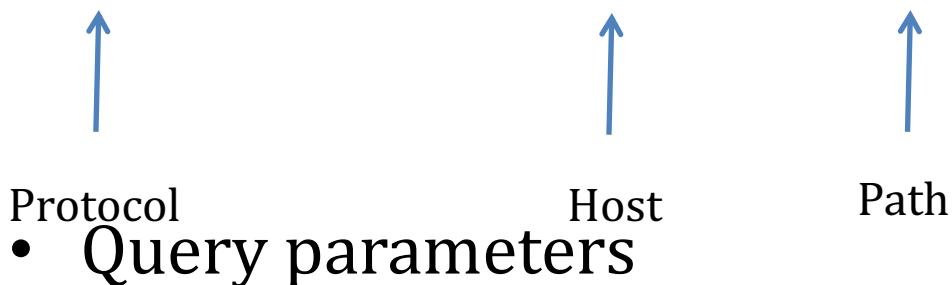
Uniform Resource Locator (URL):

[http://en.wikipedia.org/wiki/Uniform resource locator](http://en.wikipedia.org/wiki/Uniform_resource_locator)

URL

- URL: Uniform Resource Locator

`http://www.nytimes.com/pages/world/index.html`



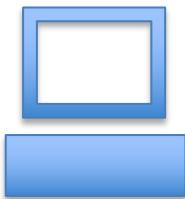
`http://www.nytimes.com/pages/world/index.html?module=HPMiniNav&contentCollection=World&pgtype=Homepage®ion=TopBar&action=click&t=qry69`

Question

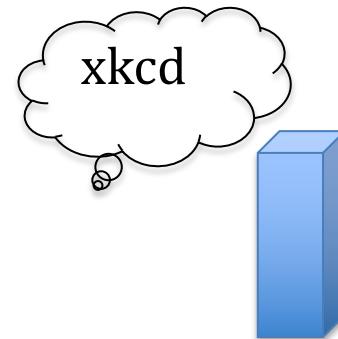
- You give the browser a URL, how does the browser know the IP address of the URL?

Question

- How does your computer know the IP address when you give only the URL?

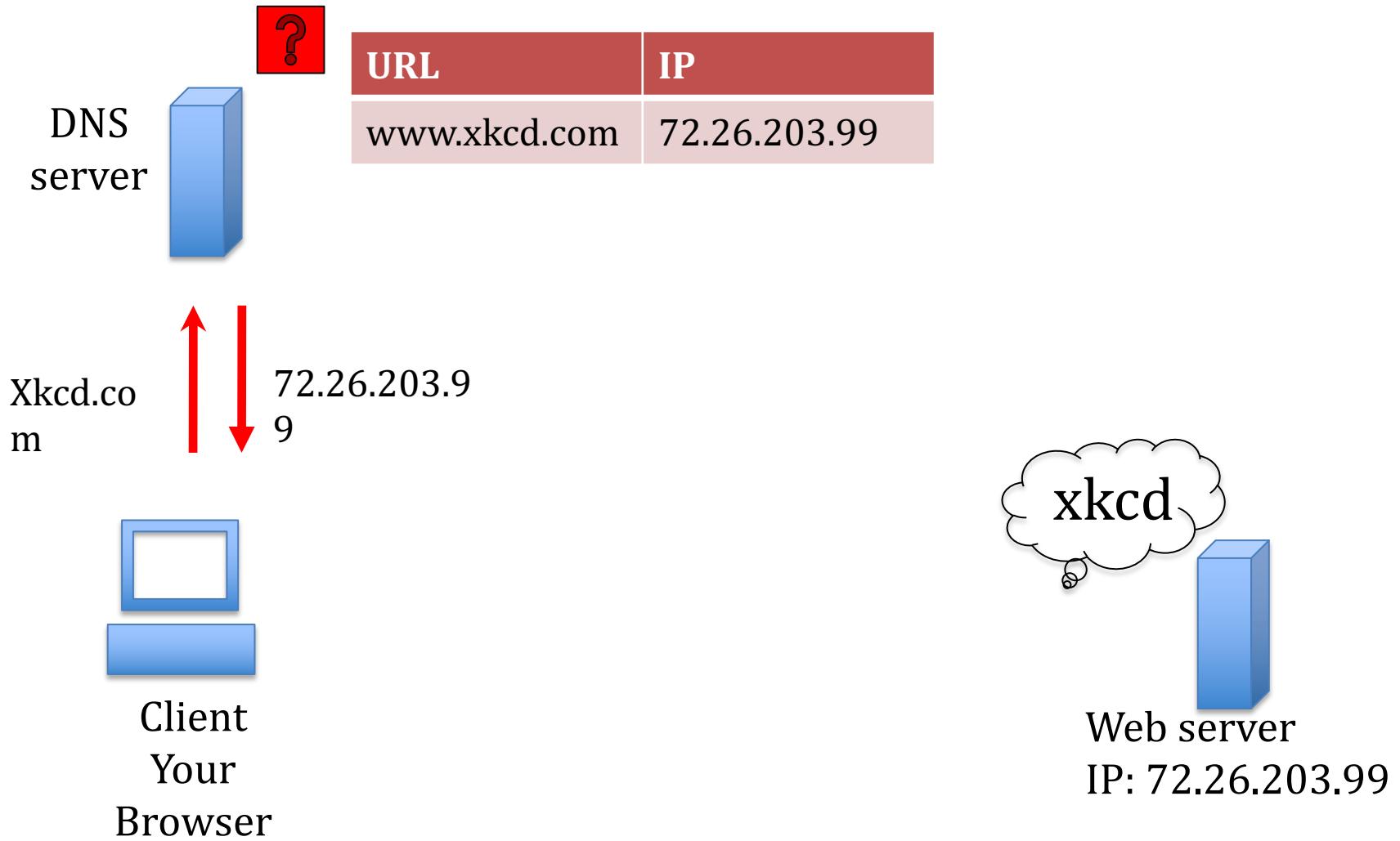


Client
Your
Browser

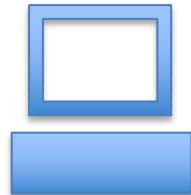


Web server
IP: 72.26.203.99

DNS (Domain Name System)

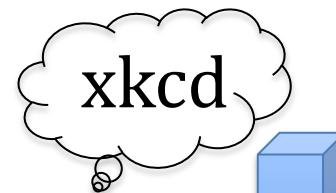


HTTP Request



Client
Your
Browser

HTTP Request: GET www.xkcd.com



Web server
IP: 72.26.203.99

Hypertext Transport Protocol (HTTP)

- The 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 request
 - PUT filename: upload

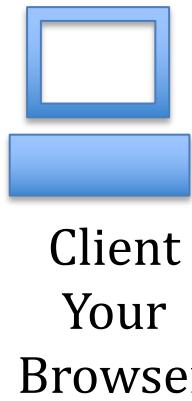
What is a HTTP Request?

- HTTP: Hyper text Transfer Protocol
<http://djce.org.uk/dumprequest>
- Your browser sends a request to the server:
Get me the file index.html

Read more here:

<http://www.jmarshall.com/easy/http/#whatis>

HTTP Response

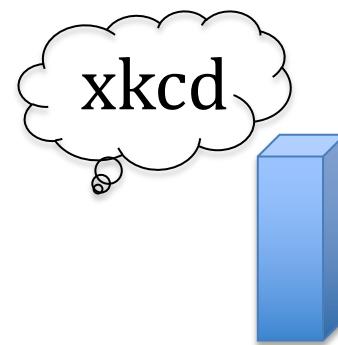


Client
Your
Browser

HTTP Request: GET www.xkcd.com



HTTP Response: web content HTML file



Web server
IP: 72.26.203.99

HTTP Error code

| Number | Meaning |
|---------|---------------------------------------|
| 200 | OK |
| 301-303 | Page has moved |
| 403 | You are forbidden to access this page |
| 404 | Page not found |
| 418 | I 'm a teapot |
| 500 | Internal server error |

[Complete List](#)

Static versus Dynamic server-side processing

- Static website: www.xkdc.com

Every user views the SAME page. So the content can be just saved on the server. In much the same way that your personal computer saves files on its hard drive.

- Dynamic website: www.facebook.com

The specific content that a user sees (the profile walls (which will change from time to time and from user to user) must be generated in real-time processing.

HTTP Response Format

HTTP/1.1 200 OK

Status Line

Date: Thu, 20 May 2004 21:12:58 GMT

General Headers

Connection: close

Server: Apache/1.3.27

Response Headers

Accept-Ranges: bytes

Content-Type: text/html

Entity Headers

Content-Length: 170

Last-Modified: Tue, 18 May 2004 10:14:49 GMT

```
<html>
<head>
<title>Welcome to the Amazing Site!</title>
</head>
```

```
<body>
<p>This site is under construction. Please come
back later. Sorry!</p>
</body>
</html>
```

Message Body

Review

- Browser map an URL into IP through DNS look up
- Send a message to web server to request to view certain file: HTTP request
- The webserver then do some server side processing and then send back the files back to the browser: HTTP responses. Fetch the files from a data server or generate the files on the fly.
- The browser decide how to display the files on the webpages.

Internet Media (“MIME”) Types

- Sometimes when including resources in a page(style sheet, icon, multimedia object), we specify their type of data

| MIME type | File extension |
|-----------------|----------------|
| Text/html | .html |
| Text/plain | .txt |
| Image/gif | .gif |
| Video/quicktime | .mov |

[List of MIME types by extension](#)

Client-side technologies

The following types of files are kind that the server can send to your computer so your browser can render the web-page:

- HTML
- CSS
- JavaScript

Client-side: HTML

- HTML describes the content of a webpage
- Headers
- Text
- Images and Videos
- Hyperlinks

Client-side: HTML

- HTML describes the content of a webpage

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4
5  </head>
6
7  <body>
8
9  <h1>CSS example!</h1>
10 <p>This is a paragraph.</p>
11
12 </body>
13 </html>
```

JavaScript

- Adds Interactivity to web-pages
- Submit forms
- Button Presses
- React to mouse movement
- Change HTML content
- Example: <http://3d.xkcd.com/?teapot>

Quiz

- What type of files can be found on the web:
 1. Music
 2. Images
 3. HTML
 4. Plain texts
 5. Videos

Server-side technology: e.g. Python CGI, PHP, Node.js

- Used to run scripts on the server, in response to HTTP requests.
- Collect form data
- Generate dynamic page content
- Handle user sessions and credentials
- Interacts with databases

Server-side technology: Database

- Database: an organized collection of data that is maintained on the server
 - Account information and passwords
 - User-generated content (blog posts, comments, clicks on facebook, etc.)
- MySQL is particular database that is commonly used with PHP.
- Queries are requests to the database for information, e.g. last five comments on a blog.
- MySQL process the queries.

PHP and MySQL

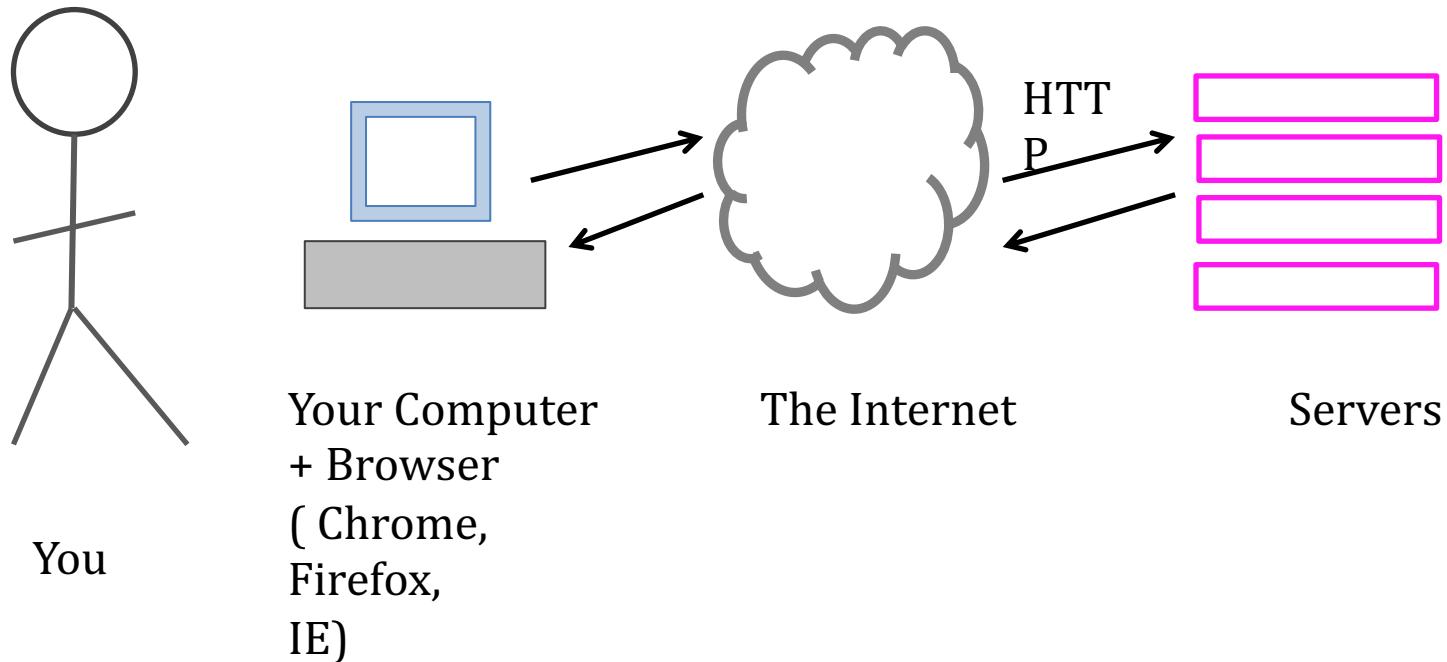
- Often work together
- PHP make queries to MySQL
- MySQL provides dynamic content
- PHP uses the content to generate a dynamic webpage.

Additional technologies

- Python CGI: The Common Gateway Interface is a standard for external gateway programs to interface with information servers such as HTTP servers.
- jQuery: JavaScript library. Very convenient syntax.
- Ajax, a group of jQuery allows you to communicate with the server without loading the pages

Example: Microsoft bing search

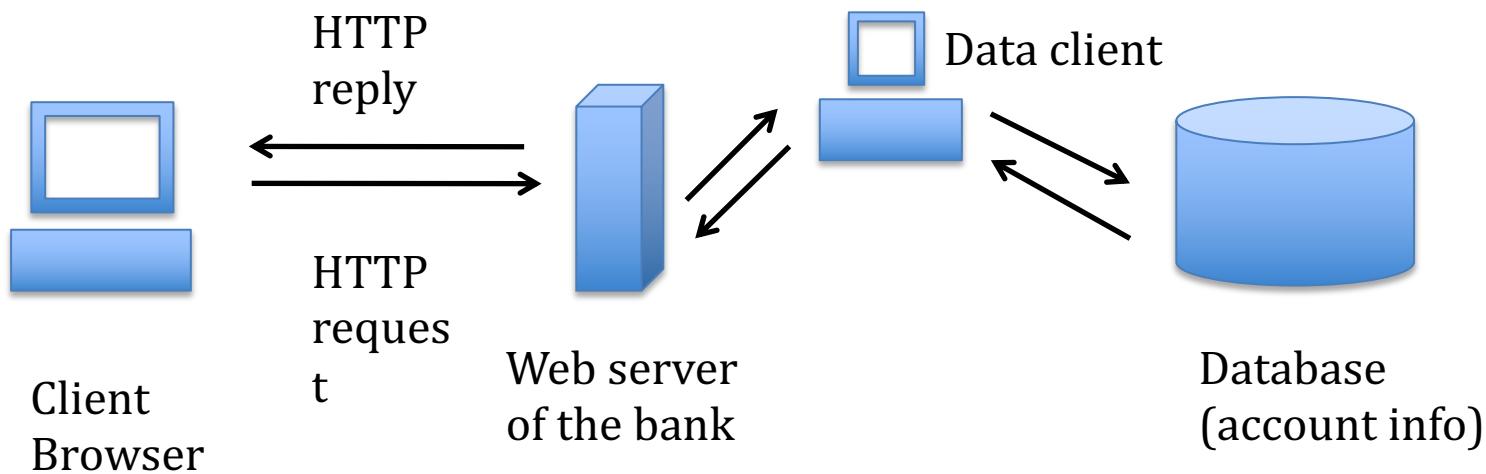
Review: Major pieces of the internet



Example: online banking

- A client access to a web browser (**the client**): www.bankofamerica.com
- The login credentials are stored in the **database**
- The **web server** access the **database server** (e.g. MySQL)
- The database server might fetch financial transaction records from another database server.
- An **application server** interprets the returned data by following the bank's business logic and return to the web server.
- The web server sends the result back to the browser, which interprets the data.

Checking account balance:



We will learn

Before Mid-term

- HTML
- CSS
- JavaScript

After Mid-term

- PythonCGI
- PHP
- MySQL
- Ajax (if time allows)
- Web frameworks

Quiz

Which of the following process the request to return web content:

1. A DNS server
2. A Web server
3. Your browser
4. A particular web server
5. All of above

Answer

Which of the following process the request to return web content:

1. A DNS server
2. A Web server
3. Your browser
4. A particular web server
5. All of above

Homework

Install one of the following editors:

<http://brackets.io>

Create an account and repository at GitHub and learn Git:

<https://github.com>

<https://guides.github.com/introduction/git-handbook/>

Please read about HTTP:

<http://www.jmarshall.com/easy/http/#whatis>

Further Readings

URL: http://en.wikipedia.org/wiki/Uniform_resource_locator

HTTP Request:

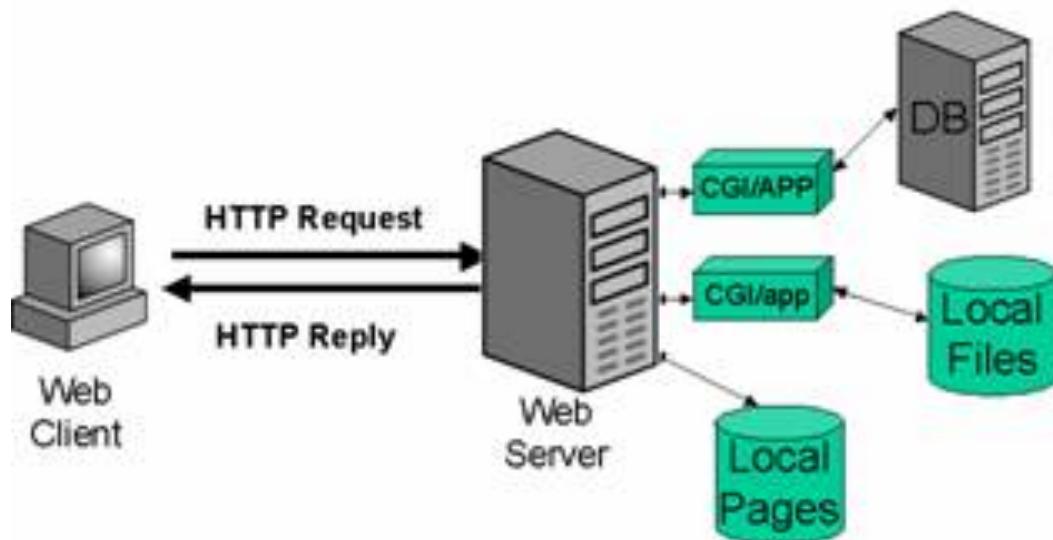
<http://djce.org.uk/dumprequest>

HTTP Response:

http://www.tcpipguide.com/free/t_HTTPResponseMessageFormat.htm

Client-Server Architecture

Web server Architecture

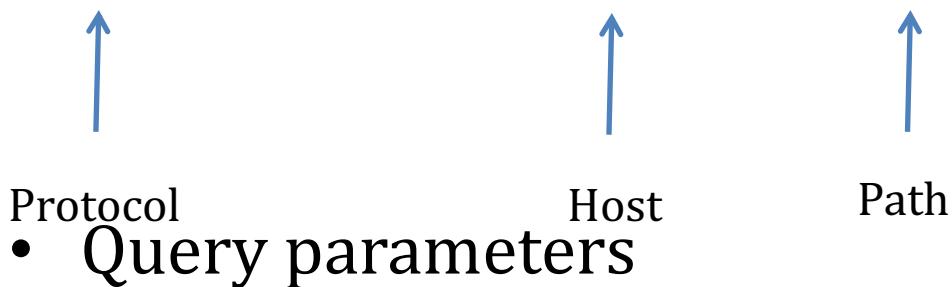


<http://www2.sans.org/security-resources/malwarefaq/guestbook.php>

URL

- URL: Uniform Resource Locator

`http://www.nytimes.com/pages/world/index.html`



`http://www.nytimes.com/pages/world/index.html?module=HPMiniNav&contentCollection=World&pgtype=Homepage®ion=TopBar&action=click&t=qry69`

Quiz

Given the following URL, identify different components of the URL

[http://www.lumosity.com/landing pages/431?gclid=CI2Wntii8LsCFUpnOgodR1UA5A](http://www.lumosity.com/landing_pages/431?gclid=CI2Wntii8LsCFUpnOgodR1UA5A)

1. Protocol
2. Host
3. Path
4. Query

Quiz

Given the protocol https, the host
www.washingtonpost.com, the path
/politics/

What is the URL?

HTTP: how browser talks to the server

- HyperText Transfer Protocol
- HTTP Request methods: Post and Get
- Get: get a document from the server

`http://www.example.com/foo`

`Get/ foo HTTP/1.1`

`Method, Path, Version`

HTTP: Get method

- Example: Amazon search for French Press returns the following URL. What is the request line to GET this URL using HTTP/1.1?

http://www.amazon.com/s/ref=nb_sb_noss_2?url=search-alias%3Daps&field-keywords=frensh%20press

HTTP Responses

- Example: Amazon search for French Press returns the following URL. What is the request line to GET this URL using HTTP/1.1?

http://www.amazon.com/s/ref=nb_sb_noss_2?url=search-alias%3Daps&field-keywords=frensh%20press