

IT4409: Web Technologies and e-Services

Term 2020-2

Instructor: Dr. Thanh-Chung Dao

Slides by Dr. Binh Minh Nguyen

Department of Information Systems
School of Information and Communication Technology
Hanoi University of Science and Technology

1

1

Reasonable Questions

- What is the World Wide Web?
- Is it the same thing as the Internet?
- Who invented it?
- How old is it?
- How does it work?
- What kinds of things can it do?
- What does it have to do with programming?

2

2

Web ≠ Internet

- **Internet** : a physical network connecting millions of computers using the same protocols for sharing/transmitting information (TCP/IP)
 - in reality, the Internet is a network of smaller networks
- **World Wide Web**: a collection of interlinked multimedia documents that are stored on the Internet and accessed using a common protocol (HTTP)

Key distinction: Internet is hardware; Web is software along with data, documents, and other media

Many other Internet-based applications exist

e.g., email, telnet, ftp, usenet, instant messaging services, file-sharing services, ...

3

3

(A Very Brief) History of the Internet

- the idea of a long-distance computer network traces back to early 60's
 - Joseph Licklider at M.I.T. (a “time-sharing network of computers”)
 - Paul Baran at Rand (tasked with designing a “survivable” communications system that could maintain communication between end points even after damage from a nuclear attack)
 - Donald Davies at National Physics Laboratory in U.K.
- in particular, the US Department of Defense was interested in the development of distributed, decentralized networks
 - survivability (i.e., network still functions despite a local attack)
 - fault-tolerance (i.e., network still functions despite local failure)

contrast with phone system, electrical system which are highly centralized services

4

4

The Internet

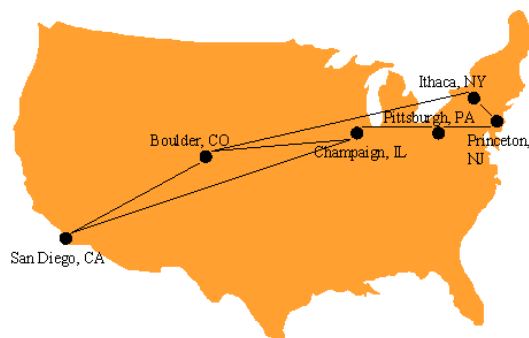
- In 1969, Advanced Research Project Agency funded the ARPANET
 - connected computers at UC Los Angeles, UC Santa Barbara, Stanford Research Institute, and University of Utah
 - allowed researchers to share data, communicate
56Kb/sec communication lines (vs. 110 b/sec over phone lines)
- Technical origin
 - One of earliest attempts to network heterogeneous, geographically dispersed computers
 - Email first available on ARPANET in 1972 (and quickly very popular!)

5

5

The Internet

- Open-access networks
 - Regional university networks (e.g., SURAnet)
 - CSNET for CS departments not on ARPANET
- NSFNET (1985-1995)
 - Primary purpose: connect supercomputer centers
 - Secondary purpose: provide backbone to connect regional networks



The 6 supercomputer centers connected by the early NSFNET backbone

6

6

Internet Growth

- throughout the 70's, the size of the ARPANET doubled every year
 - first ARPANET e-mail sent in 1971
 - decentralization made adding new computers easy
 - TCP/IP developed in the mid 1970s for more efficient packet routing
 - migration of ARPANET to TCP/IP completed 1 January, 1983
 - ~1000 military & academic host computers connected by 1984
- in 80's, U.S. government took a larger role in Internet development
 - created NSFNET for academic research in 1986
 - ARPANET was retained for military & government computers
- by 90's, Internet connected virtually all colleges & universities
 - businesses and individuals also connecting as computing costs fell
 - ~1,000,000 computers by 1992
- in 1992, control of the Internet was transferred to a non-profit organizations
 - Internet Society:
 - Internet Engineering Task Force
 - Internet Architecture Board
 - Internet Assigned Number Authority
 - World-Wide-Web Consortium (W3C)

7

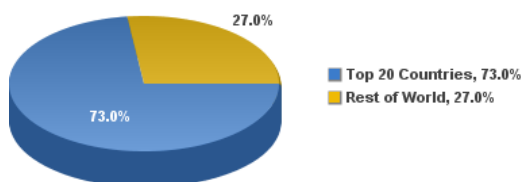
7

Internet Growth (cont.)

Internet has exhibited exponential growth,
doubling in size every 1-2 years
(stats from *Internet Software Consortium*)

United Kingdom has 52.7 million users (approx.
83.6% of the population)

**Top 20 Internet Countries
versus World - November 30, 2015**



Source: Internet World Stats - www.internetworldstats.com/top20.htm
3,366,261,156 estimated Internet users for November 30, 2015
Copyright © 2016, Miniwatts Marketing Group

Year	Computers on the Internet (at any one time?)
2011	~605,000,000
2006	439,286,364
2004	285,139,107
2002	162,128,493
2000	93,047,785
1998	36,739,000
1996	12,881,000
1994	3,212,000
1992	992,000
1990	313,000
1988	56,000
1986	5,089
1984	1,024
1982	235

8

8

Internet users in Vietnam



From dammio

9

9

(A Very Brief) History of the Web

- the idea of hypertext (cross-linked and inter-linked documents) traces back to Vannevar Bush in the 1940's
 - online hypertext systems began to be developed in 1960's
 - e.g., Ted Nelson and Andy van Dam's Hypertext Editing System (HES), Doug Englebert's NLS (on-Line System)
 - in 1987, Apple introduced HyperCard (a hypermedia system that predated the WWW)
- in 1989, Tim Berners-Lee at the European Particle Physics Laboratory (CERN) designed a hypertext system for linking documents over the Internet
 - designed a (Non-WYSIWYG) language for specifying document content
 - evolved into HyperText Markup Language (HTML)
 - designed a protocol for downloading documents and interpreting the content
 - evolved into HyperText Transfer Protocol (HTTP)
 - implemented the first browser -- text-based, no embedded media

the Web was born!

10

10

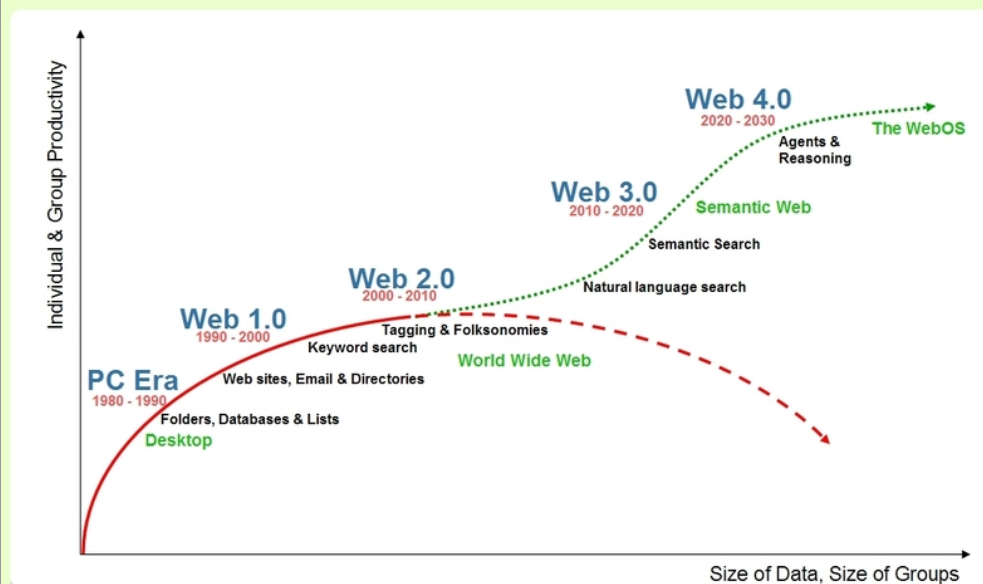
History of the Web (cont.)

- the Web was an obscure, European research tool until 1993
- in 1993, Marc Andreessen and Eric Bina (at the National Center for Supercomputing Applications, a unit of the University of Illinois) developed **Mosaic**, one of the early graphical Web browsers that popularized the WWW for the general public (Erwise was the first one, ViolaWWW the second)
 - the intuitive, clickable interface helped make hypertext accessible to the masses
 - made the integration of multimedia (images, video, sound, ...) much easier
 - Andreessen left NCSA to found Netscape in 1994
 - cheap/free browser further popularized the Web (75% market share in 1996)
- in 1995, Microsoft came out with Internet Explorer
- Opera web browser released in 1996
 - Netscape bought by AOL in 1998 for US\$4.2 billion in stock
 - Firefox web browser, version 1.0, released in 2004
 - Google Chrome released in 2008
- today, the Web is the most visible aspect of the Internet

11

11

The Future of Productivity



Source: Radar Networks & Nova Spivack, 2007 – www.radarnetworks.com

12

12

Popular websites in Vietnam

#	WEBSITE	CATEGORY	MONTHLY TRAFFIC	TIME PER VISIT	PAGES PER VISIT
01	GOOGLE.COM.VN	SEARCH	375,300,000	12M 26S	6.8
02	FACEBOOK.COM	SOCIAL	351,400,000	25M 04S	21.7
03	YOUTUBE.COM	TV & VIDEO	329,900,000	27M 15S	11.9
04	GOOGLE.COM	SEARCH	323,200,000	13M 24S	15.3
05	VNEXPRESS.NET	NEWS & MEDIA	73,400,000	9M 07S	4.2
06	ZING.VN	SOCIAL	61,900,000	14M 47S	6.6
07	YAHOO.COM	NEWS & MEDIA	41,200,000	5M 49S	4.6
08	COCCOC.COM	SEARCH	37,500,000	6M 54S	2.6
09	NEWS.ZING.VN	NEWS & MEDIA	32,400,000	7M 11S	3.6
10	KENH14.VN	NEWS & MEDIA	32,100,000	7M 31S	4.8

From dammio

13

13

World Wide Web

- **The Web** is the collection of machines (Web servers) on the Internet that provide information, particularly HTML documents, via HTTP.
- Machines that access information on the Web are known as **Web clients**. A **Web browser** is software used by an end user to access the Web.

14

14

Hypertext Transport Protocol (HTTP)

- HTTP is based on the request-response communication model:
 - Client sends a request
 - Server sends a response
- HTTP is a stateless protocol:
 - The protocol does not require the server to remember anything about the client between requests.

15

15

HTTP

- Normally implemented over a TCP connection (80 is standard port number for HTTP)
- Typical browser-server interaction:
 - User enters Web address in browser
 - Browser uses DNS to locate IP address
 - Browser opens TCP connection to server
 - Browser sends HTTP request over connection
 - Server sends HTTP response to browser over connection
 - Browser displays body of response in the client area of the browser window

16

16

HTTP Request

Structure of the request:

- start line
- header field(s)
- blank line
- optional body

17

17

HTTP Request

Structure of the request:

- **start line**
- header field(s)
- blank line
- optional body

18

18

HTTP Request

Start line

- Example: GET / HTTP/1.1

Three space-separated parts:

- HTTP request method
- Request-URI ([Uniform Resource Identifier](#))
- HTTP version

19

19

HTTP Request

Start line

- Example: GET / HTTP/1.1

Three space-separated parts:

- HTTP request method
- Request-URI
- **HTTP version**

We will cover 1.1, in which version part of start line must be exactly as shown

20

20

HTTP Request

Start line

- Example: GET / HTTP/1.1

Three space-separated parts:

- HTTP request method
- **Request-URI**
- HTTP version

21

21

HTTP Request

- Uniform Resource Identifier (URI)

- Syntax: *scheme* : *scheme-depend-part*

Ex: in `http://www.example.com/`

the scheme is `http`

- Request-URI is the portion of the requested URI that follows the host name (which is supplied by the required Host header field)

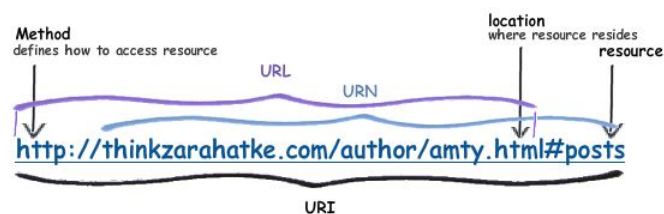
Ex: `/` is Request-URI portion of `http://www.example.com/`

22

22

URI

- URI's are of two types:
 - Uniform Resource Name (URN)
 - Can be used to identify resources with unique names, such as books (which have unique ISBN's)
 - Scheme is `urn`
 - Uniform Resource Locator (URL)
 - Specifies location at which a resource can be found
 - In addition to `http`, some other URL schemes are `https`, `ftp`, `mailto`, and `file`



23

23

HTTP Response

Structure of the response:

- status line
- header field(s)
- blank line
- optional body

24

24

HTTP Response

Structure of the response:

- **status line**
- header field(s)
- blank line
- optional body

25

25

HTTP Response

Status line

- Example: HTTP/1.1 200 OK

Three space-separated parts:

- HTTP version
- status code
- reason phrase (intended for human use)

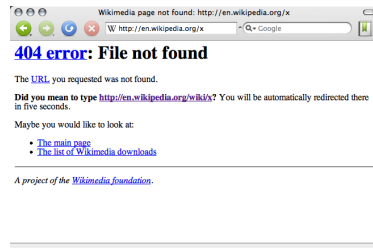
26

26

HTTP Response

Status code

- Three-digit number
- First digit is class of the status code:
 - 1=Informational
 - 2=Success
 - 3=Redirection (alternate URL is supplied)
 - 4=Client Error
 - 5=Server Error
- Other two digits provide additional information
- See <http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html>



27

27

HTTP Response

Structure of the response:

- status line
- header field(s)
- blank line
- optional body

28

28

HTTP Response

Common header fields:

- **Connection**, **Content-Type**, **Content-Length**
- **Date**: date and time at which response was generated (required)
- **Location**: alternate URI if status is redirection
- **Last-Modified**: date and time the requested resource was last modified on the server
- **Expires**: date and time after which the client's copy of the resource will be out-of-date
- **ETag**: a unique identifier for this version of the requested resource (changes if resource changes)

29

29

HTTP Request/Response Examples

```
Connect    {  $ telnet www.example.org 80
              Trying 192.0.34.166...
              Connected to www.example.com
              (192.0.34.166).
              Escape character is '^]'.

Send Request {  GET / HTTP/1.1
              Host: www.example.org

Receive Response {  HTTP/1.1 200 OK
                  Date: Thu, 09 Oct 2003
                  20:30:49 GMT
                  ...
```

30

30

Web Browsers

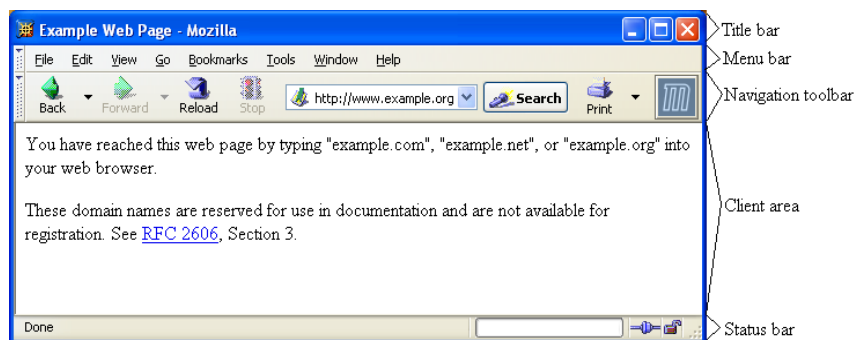
First graphical browser running on general-purpose platforms:



31

31

Web Browsers



32

32

Web Browsers

Primary tasks:

- Convert web addresses (URL's) to HTTP requests
- Communicate with web servers via HTTP
- **Render** (appropriately display) documents returned by a server

33

33

Static vs. Dynamic pages

- most Web pages are *static*
 - contents (text/links/images) are the same each time it is accessed

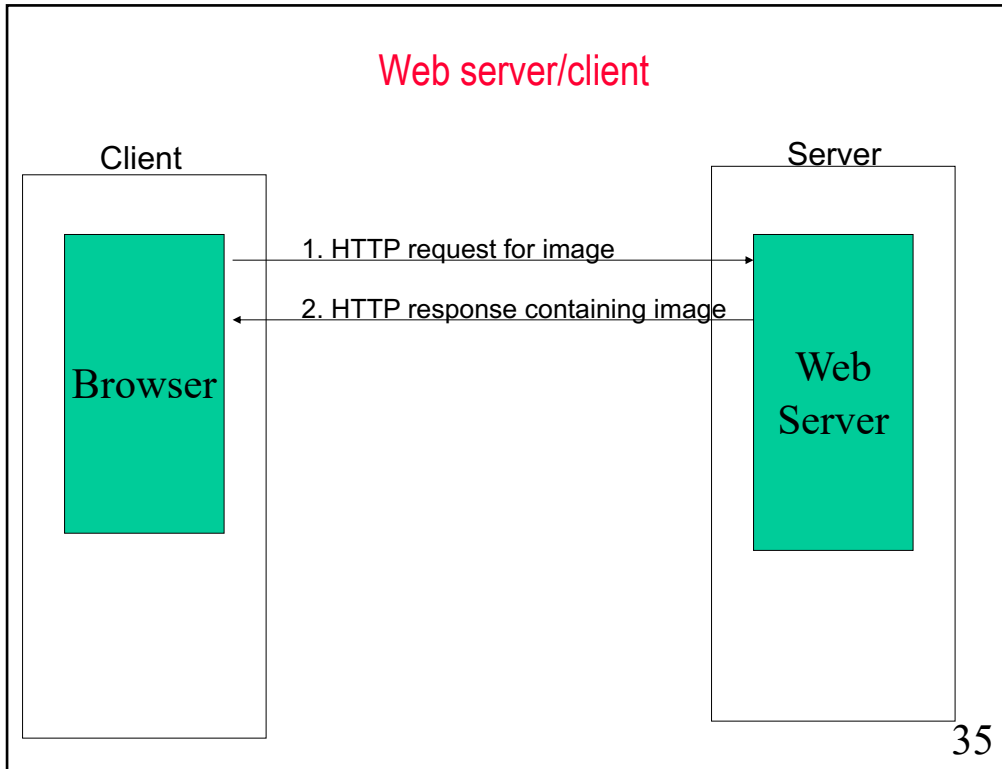
e.g., online documents, most homepages

HyperText Markup Language (HTML) is used to specify text/image format
- as the Web continues to move towards more and more online services and e-commerce continues to grow, Web pages must also provide *dynamic* content
 - pages can be fluid, changeable (e.g., rotating banners)
 - must be able to react to the user's actions, request and process info, tailor services

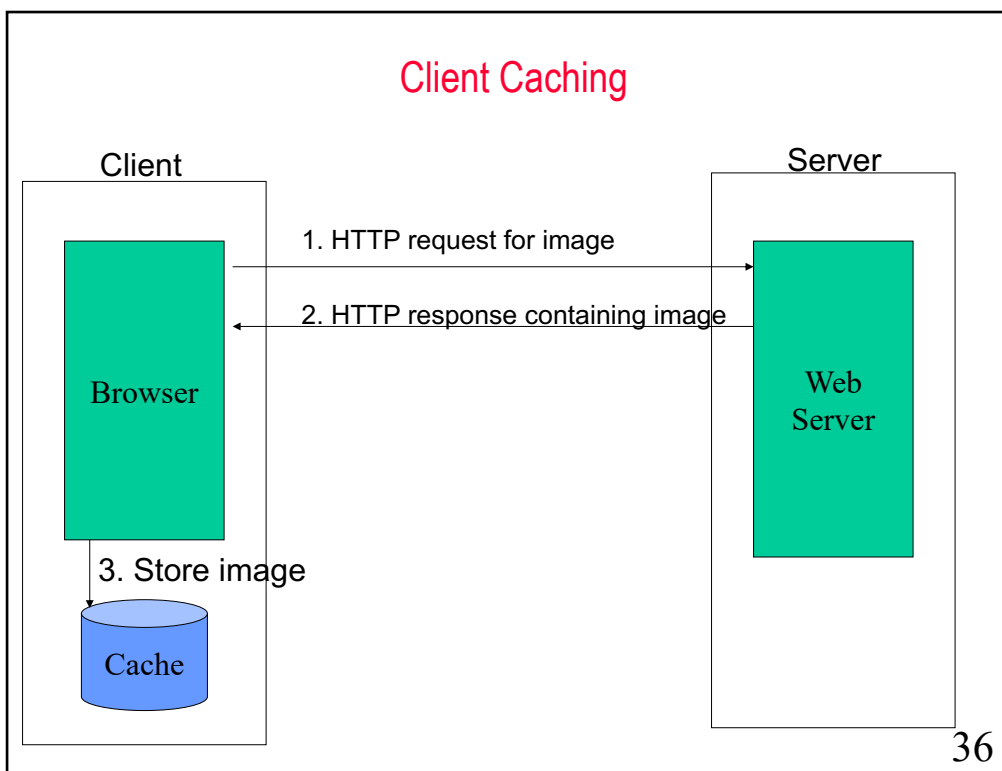
e.g., amazon.com
- this course is about applying your programming skills to the development of dynamic Web pages and applications

34

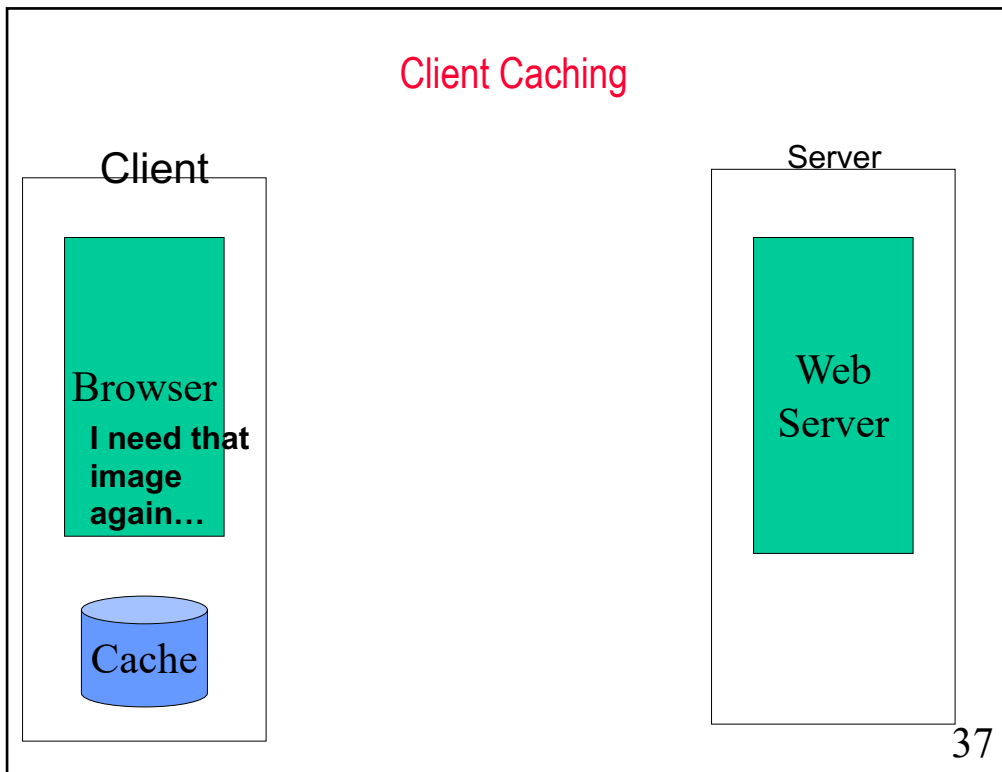
34



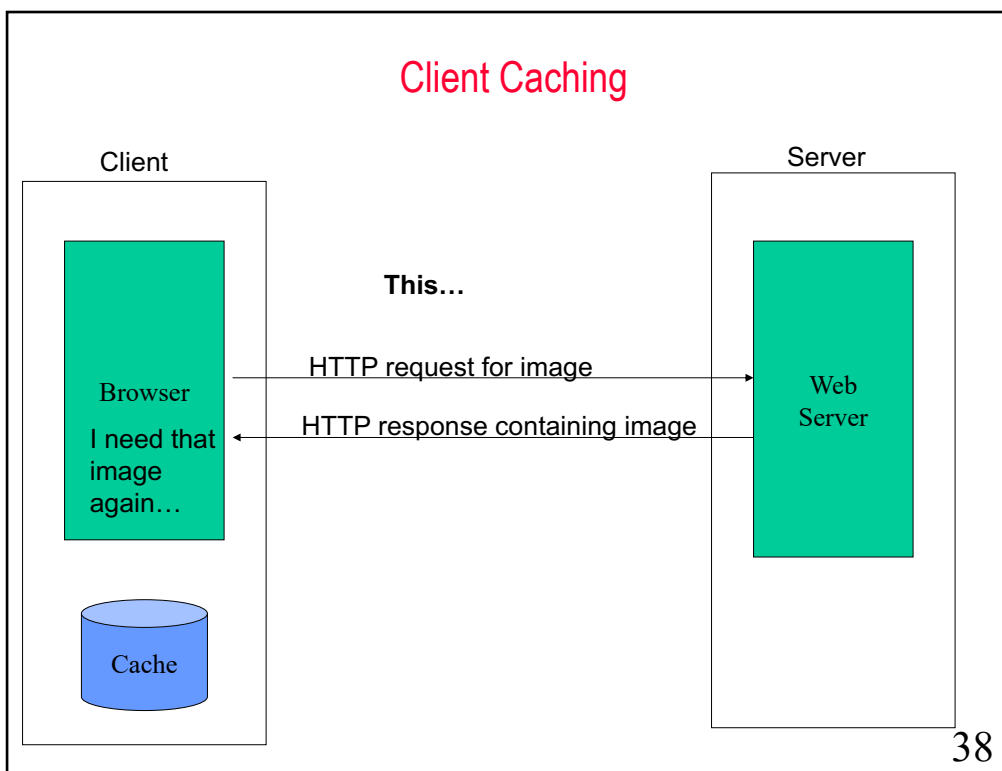
35



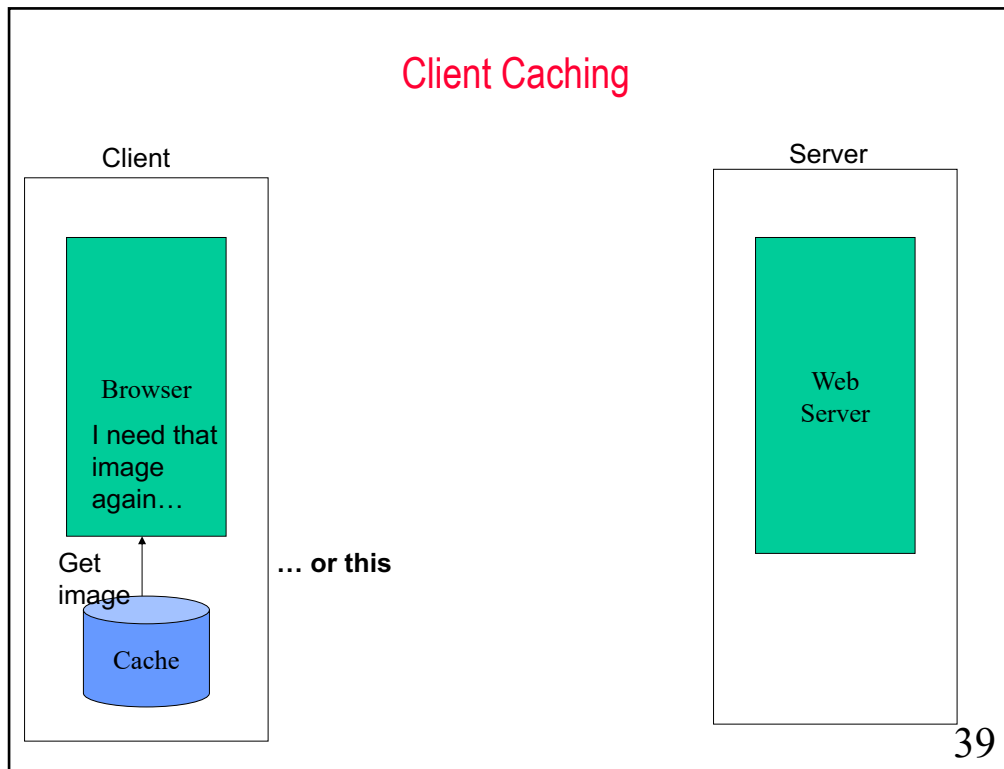
36



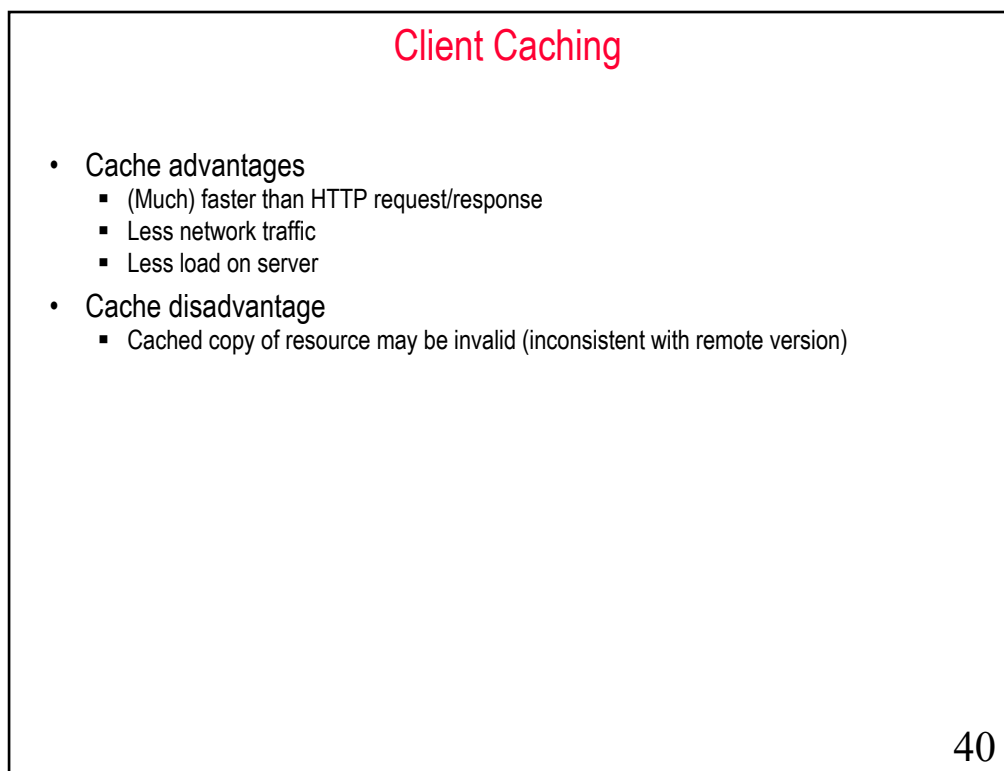
37



38



39



40

Web Clients

- Many possible web clients:
 - Text-only “browser” (lynx)
 - Mobile phones
 - Robots (software-only clients, e.g., search engine “crawlers”)
 - etc.
- We will focus on traditional web browsers

41

41

Web Servers

Basic functionality:

- Receive HTTP request via TCP
- Map host header (domain name) to specific **virtual host** (one of many host names sharing an IP address)
- Map Request-URI to specific resource associated with the virtual host
 - File: Return file in HTTP response
 - Program: Run program and return output in HTTP response
- Map type of resource to appropriate MIME type and use to set Content-Type header in HTTP response
- Log information about the request and response

42

42

Web Servers

httpd: UIUC, primary Web server c. 1995

Apache: "A patchy" version of httpd, now the most popular server (esp. on Linux platforms)

IIS: Microsoft Internet Information Server

Tomcat:

- Java-based
- Provides container (Catalina) for running Java servlets (HTML-generating programs) as back-end to Apache or IIS
- Can run stand-alone using Coyote HTTP front-end

43

43

Client-Side Programming

- can download program with Web page, execute on client machine
 - simple, generic, but sometimes insecure
- JavaScript
 - a scripting language for Web pages, developed by Netscape in 1995
 - uses a C++/Java-like syntax, so familiar to programmers, but simpler
 - good for adding dynamic features to Web page, controlling forms and GUI
 - requires users to have this technology enabled on their browsers
 - see <http://www.w3schools.com/js/>
- Java applets
 - can define small, special-purpose programs in Java called *applets*
 - provides (almost) full expressive power of Java (but with more overhead)
 - good for more complex tasks or data heavy tasks, such as graphics
 - see <http://java.sun.com/applets/>

44

44

Server-Side Programming

- can store and execute program on Web server, link from Web page
 - more complex, requires server privileges, but can still be (mostly) secure
- *Common Gateway Interface (CGI) programming*
 - programs can be written to conform to the CGI
 - when a Web page submits, data from the page is sent as input to the CGI program
 - CGI program executes on server, sends its results back to browser as a Web page
 - good if computation is large/complex or requires access to private data
- **Active Server Pages (ASP), Java Servlets, PHP, Server Side Includes, Ajax**
 - some of these are vendor-specific alternatives to CGI (such as Microsoft's ASP)
 - provide many of the same capabilities as CGI programs but using HTML-like tags
 - some of these technologies might require functionality to be enabled in the client's browser (e.g. Ajax generally requires the use of Javascript combined with PHP or some other server-based programming component)

45

45

email: chungdt@soict.hust.edu.vn

Q&A

46

46