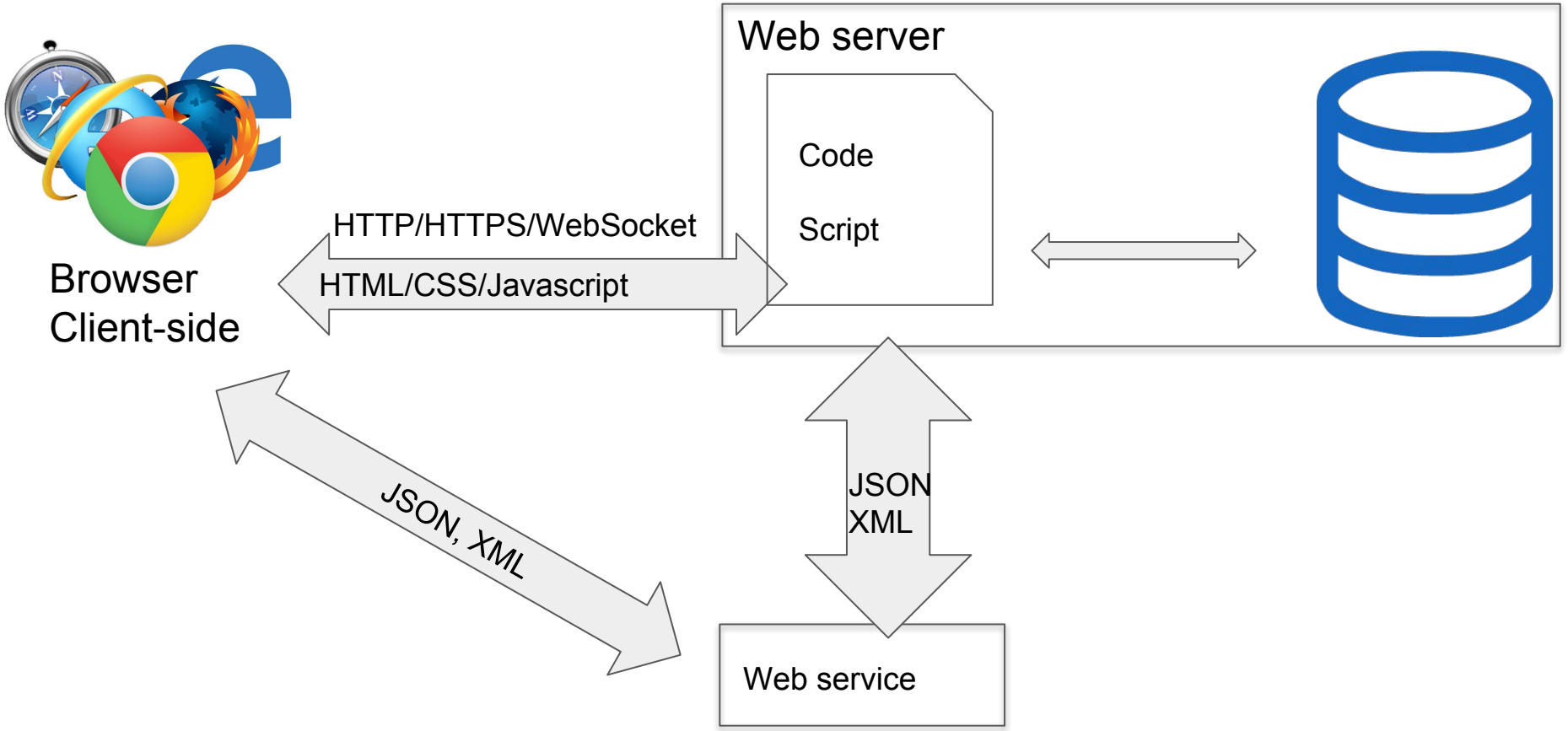
CP 476
Internet Computing

Instructor: Masoomeh Rudafshani

Agenda

- WWW
  - Web Browsers
  - Web Servers
  - URL
  - HTML
  - MIME
  - HTTP
- Software tools
- Course page has moved to:
  - https://bohr.wlu.ca/mrudafshani/cp476/

Web-based applications



World Wide Web

- Origins
  - Tim 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
  - Non-sequential browsing of textual material
- Pages? Documents? Resources?
  - We'll call them documents
- Hypermedia
  - More than just text - images, audio, etc.

WWW: History

- The early web was all about hypertext, the H in HTML
  - Browsers used to help users find internet resources
- Wayback machine
  - A huge archive of past web content
  - https://archive.org/web/web.php
  - Transition from page bv page architecture of early websites
    - Web 2.0

Web browsers

- Early browsers had no GUI
- 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
- Browsers are clients - always initiate, servers react
  - although sometimes servers require responses
- 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

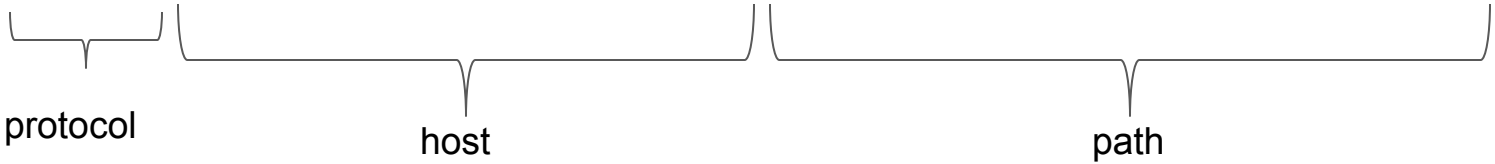
Web Browsers: usage statistics

- Chrome is the favorite browser we are working with
  - http://gs.statcounter.com/
- What is supported by a specific browser
  - https://caniuse.com/

URL (Uniform Resource Locatior)

An identifier for the location of a document

https://hopper.wlu.ca/~mrudafshani/cp476/



URL

- General form:
  - scheme:object-address
- Schem: a communications protocol: telnet, ftp, mailto, file, http
- For the http protocol, the object-address is:
- fully qualified domain name/doc path
  - Host name may include a port number, as in
  - zeppo:80 (80 is the default, so this is silly)
- For the file protocol, only the doc path is needed

URL

- URLs cannot include spaces or any of a collection of other special characters such as (semicolons, colons, ...)
  - Space are replaced with ASCII code of space: %20
- If the doc path ends with a slash, it means it is a directory

MIME (Multipurpose Internet Mail Extension)

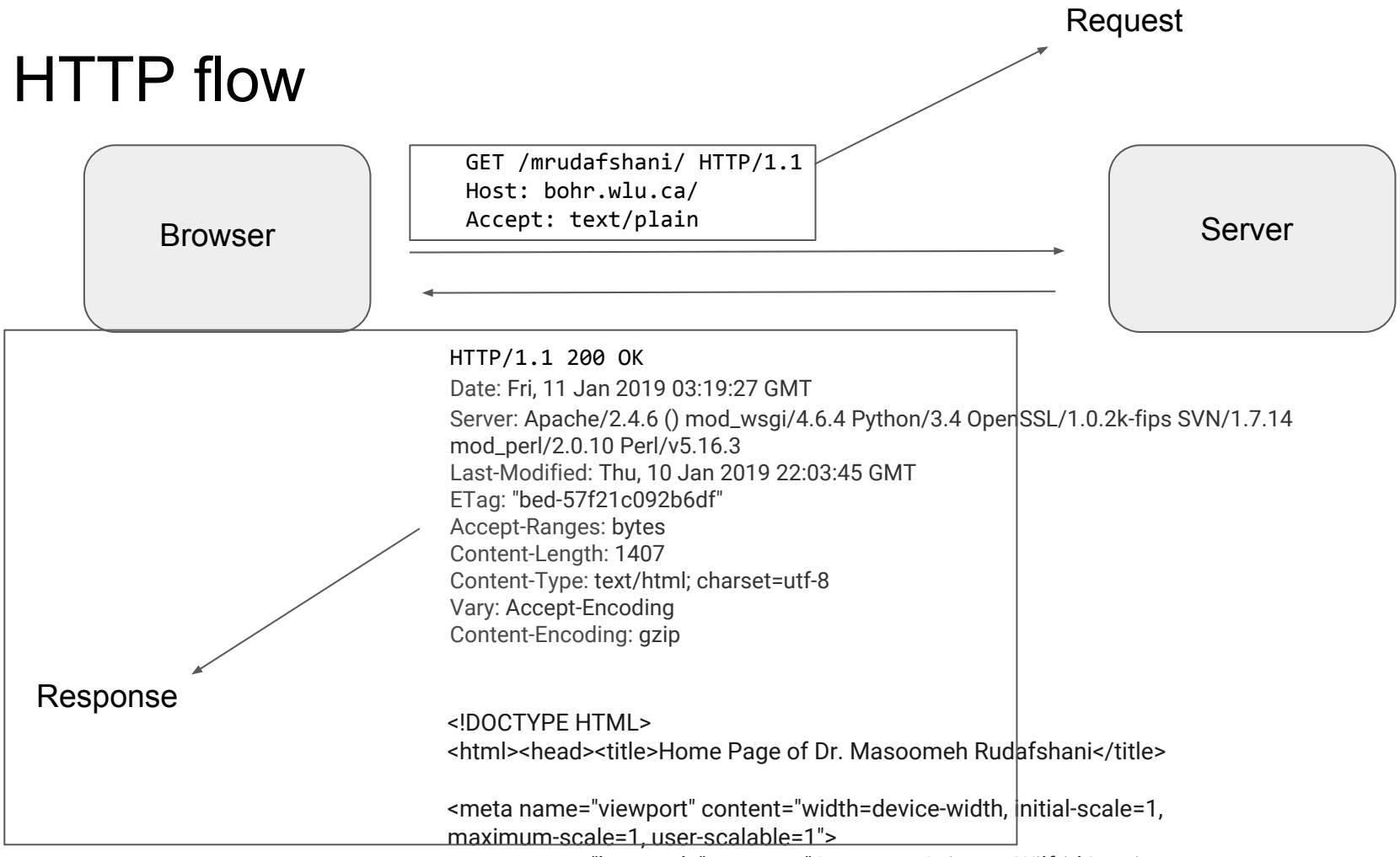
- Originally defined for email
- Used to specify to the browser the form of a file returned by the server
  - Attached by the server to the resposne
- Type specifications
  - type/subtype
    - text/plain, text/html, image/gif, image/jpeg
  - Experimental types
    - Subtype begins with x- e.g., video/x-msvideo
- Experimental types require the server to send a helper application or plug-in so the browser can deal with the file

HTTP (HyperText Transfer Protocol)

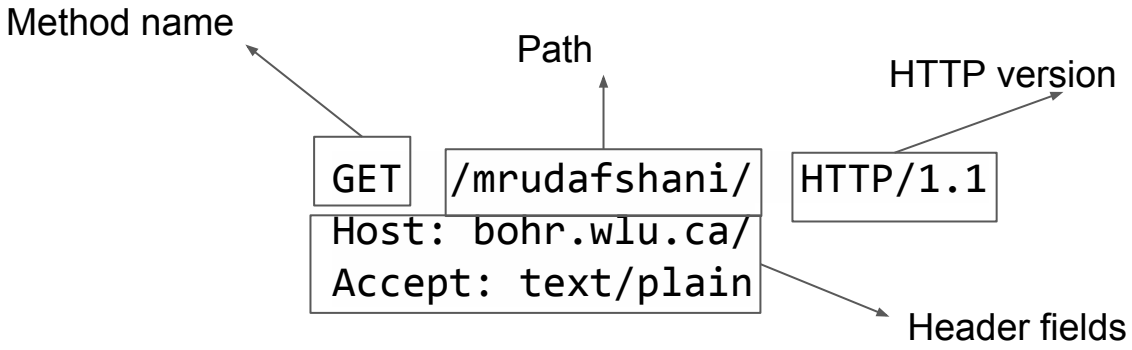
- HTTP defines how clients request web pages from the web server and how servers transfer web pages to client

- Connection establishment
  - Over TCP/IP
- Request phase
  - body/header
- Response phase
  - body/header
  -

HTTP flow


Request
Browser
GET /mrudafshani/ HTTP/1.1
Host: bohr.wlu.ca/
Accept: text/plain
Server
HTTP/1.1 200 OK
Date: Fri, 11 Jan 2019 03:19:27 GMT
Server: Apache/2.4.6 () mod_wsgi/4.6.4 Python/3.4 OpenSSL/1.0.2k-fips SVN/1.7.14
mod_perl/2.0.10 Perl/v5.16.3
Last-Modified: Thu, 10 Jan 2019 22:03:45 GMT
ETag: "bed-57f21c092b6df"
Accept-Ranges: bytes
Content-Length: 1407
Content-Type: text/html; charset=utf-8
Vary: Accept-Encoding
Content-Encoding: gzip
Response
<!DOCTYPE HTML>
<html><head><title>Home Page of Dr. Masoomeh Rudafshani</title>
<meta name="viewport" content="width=device-width, initial-scale=1,
maximum-scale=1, user-scalable=1">

HTTP Request

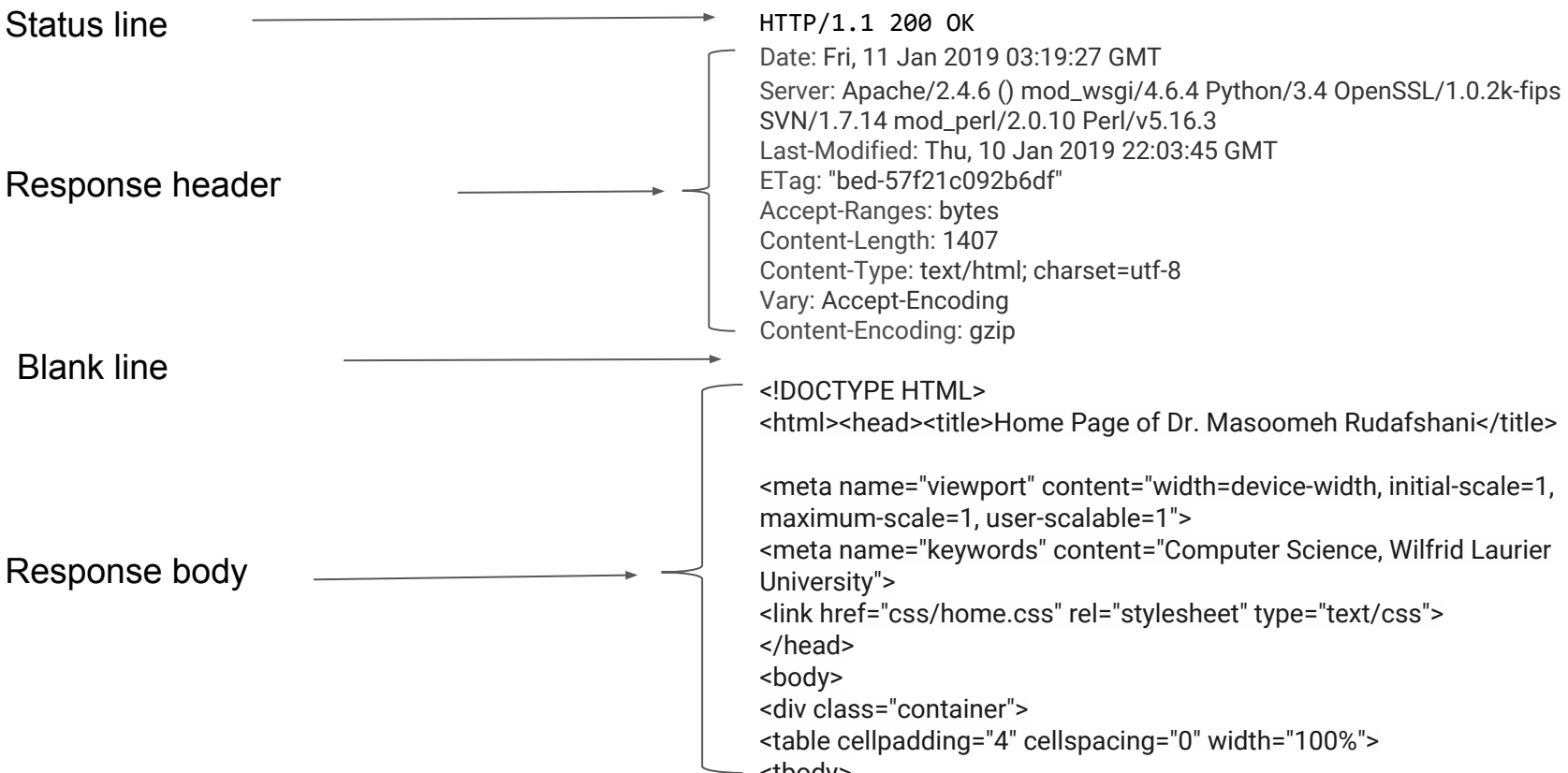


- General format of a request:
  - Method name path of the URL HTTP Version
    - Most commonly used methods
      - Get, Post, Head, Fetch, PUT, Delete
  - Header fields
    - Field name: value
  - Blank line
  - Message body

HTTP Request Methods

- GET
  - Returns the content of a specified document
- HEAD
  - Returns the header information for a specified document
- POST
  - Send form data from a browser to a server
  - A request to execute a server-side program that will process the data
- PUT
  - Replaces a specified document with the enclosed data
- DELTE
  - Deletes a specified document

HTTP Response



HTTP Response: Status line

Http version	Status code	Status phrase

- Http Version
- Status code
  - 5 groups of three digits integers indicating the result of the attempt to satisfy the request

- Status phrase

HTTP Response: Status line

Http version	Status code	Status phrase

- Status code:
  - 1xx: Informational
  - 2xx: Success
  - 3xx: Redirection
  - 4xx: Client Error
  - 5xx: Server Error
- Status phrase

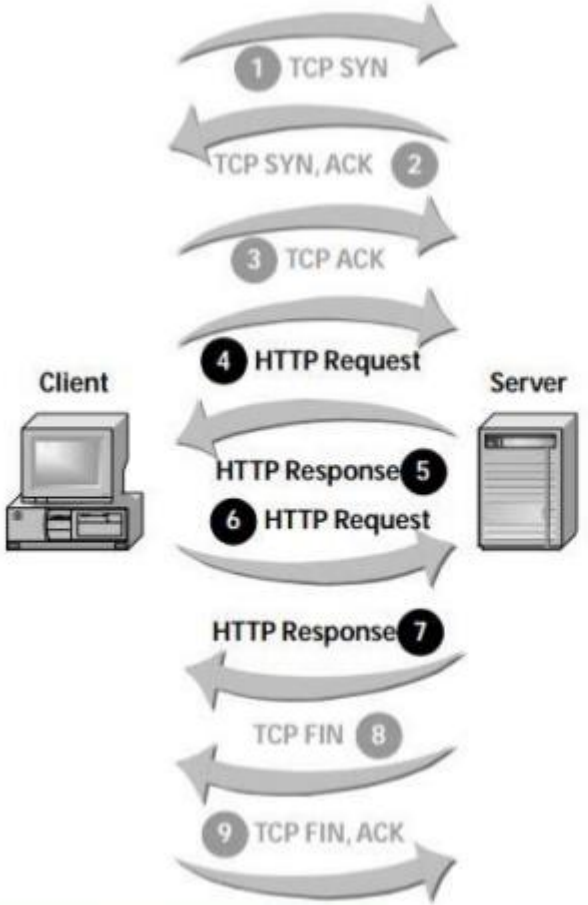
- Example Status code:
  - 404
    - Not Found
  - 200
    - OK
  - 500
    - Server Error

HTTP (HyperText Transfer Protocol)

- HTTP Versions
  - HTTP 1.0
    - https://tools.ietf.org/html/rfc1945
    - Non-persistent: the connection is closed after a request is done
  - HTTP 1.1 is defined as RFC 2616
    - https://datatracker.ietf.org/doc/rfc2616/
      - It has many updates
    - Persistent: the connection will keep alive for a short period of time
    - Another request coming within the time can use the same connection
  - HTTP 2.0
    - https://tools.ietf.org/html/rfc7540
    - encryption
-

HTTP (connection establishment)

- Persistent conneciton
  - Many http messages over the same TCP conneciton



Taken from [2]

HTTP Headers: General Headers

- For general information such as date
  - Connection
    - Lets clients and server manage connection state
    - Connection: keep=alive
    - Connection: close
  - Date
    - When the message was created
  - Via
    - Shows proxies that handled message
  - Cache-control
    - Enables caching directives

Request headers

- Host
  - The hostname of server to which request is being sent
- Referrer
  - The URL of the resource from which the current request URL came
- User-Agent
  - Name of requesting applications
- Accept
  - Inform user of client’s acceptable mime types
- Cookie
  - Pass cookies back to the server by client

Response Headers

- Server
  - The server name and version
- Set-cookie
  - How a server sets a cookie on a client

HTTP

- Stateless
  - Requests are not connected
  - HTTP has no memory of you making the first request
- Solution:
  - Session
    - Passing information back and forth in the form of a cookie
    - Cookies are sent in http headers

Cookies

- Small piece of data that a server sends to the user’s web browser
-
- Creating cookies
  - Response header
  - Set-cookie: <cookie-name>=<cookie-value>
- The browser will include the cookie in the next requests to the serve
  - Cookie header
  - cookie: <cookie-name>=<cookie-value>

HTTPS

In order to add security to HTTP, Netscape created HTTPs in 1994

HTTPS = HTTP over SSL (and later over TSL)

Usually served on port 443

Web Servers

- Provide responses to browser requests
  - existing documents or dynamically built documents
- All communications use HTTP
- Web servers run as background processes in the operating system
  - Monitor a communications port on the host
  - accepting HTTP messages when they appear
  - Perform operations specified by the http message
    - URL: include the specification of a host server machine
    - URL is translated into a file name and the file is returned to the client
    - URL is translated into a program name (the program is run and its output is sent to the requesting client

Web Servers Characteristics

- Web servers have two main directories:
  - Document root (servable documents)
    - Stores the web documents
    - Document root is accessed indirectly by clients
      - Server maps requested URL to the files in this directory
        - Its location is not known to the user
  - Server root (server system software)
    - Its actual location is set by the server configuration file

Web Servers Characteristics

- Virtual document trees
  - Server is configured to direct-request URLs with a particular file path to a storage area separate from the document-root directory.
  - Image, for example, are usually located on a separate directory/server
- Virtual hosts
  - Support for more than one site on a computer
    - Reducing the cost of each site
    - Making their maintenance more convenient

Web Servers

bohr.wlu.ca/~mrudafshani

Document root: public_html

bohr.wlu.ca

/var/www

Proxy Server

- A server that receives requests intended for another server and that acts on the behalf of the client to obtain the requested service
- Http client sends a request to the proxy server
- The proxy server sends the request back
- A proxy server can keep copies of responses to recent requests for further requests from other clients
- A proxy server is often used when the client and the server are incompatible for direct connection
- It may also be used for screening purposes to enable the administrator to control access to undesirable sites

Web Servers

- Proxy servers
- Two most common server configuration
  - Apache on linux
  - IIS on windows
- Apache (open source, fast, reliable)
  - Began as the NCSA server, httpd
  - Maintained by editing its configuration file
  - free and open-source cross-platform web server
- Microsoft IIS
  - Is supplied as part of Windows
-

Web Servers


Apache
Nginx
Microsoft-IIS
LiteSpeed
Google Servers
80
70
60
50
40
30
20
10
0
1 Jan'10
1 Jan'11
1 Jan'12
1 Jan'13
1 Jan'14
1 Jan'15
1 Jan'16
1 Jan'17
1 Jan'18
1 Jan'19
Usage of web servers for websites, 11 Jan 2019, W3Techs.com

HTML: Hypertext Markup Language

- Purpose
  - To describe the general form and layout of documents
- An HTML document is a mix of content and controls
  - Controls
    - 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
    - <img src = “redhead.jpg”>

HTML: Hypertext Markup Language

Tools for creating HTML Documents

- HTML Editors
  - make document creation easier
    - Shortcuts to typing tag names, spell-checker,
    - Eclipse (IDE)
      - Web Support
    - Brackets (IDE)
      - http://brackets.io/
- WYSIWYG (What you see is what you get) HTML Editors
  - Microsoft's FrontPage
  - Online editors: https://html-online.com/editor/
  - Google doc html editor: http://htmleditor.kwebpia.net/

How to see the HTTP messages

- Developer Tools
- Rest Client
  - An extension for chrome browser

Environment Setup

- Test environment
  - Hopper.wlu.ca
    - Document root: PUBLIC_HTML
  - How to connect to hopper
    - SSh, PUTTY
    - https://bohr.wlu.ca/cp367/labs/lab01_introduction.php?d=2019-01-13
-

TEST and Development environment

- First option
  - Text editor/IDE
    - Brackets (http://brackets.io/)
    - Eclipse
  - FileZilla (move the files to hopper)
- Second option
  - Eclipse Remote System Explorer
- Third option
  - Text editor/IDE
  - Local test environment
    - XAMPP
    - FileZilla (move files to hopper)

Work on a remote project with Eclipse via SSH

- It is needed to instal Remote System Explorer (RSE)
  - a set of plug-ins to work on a remote project with Eclipse via SSH
- To check if RSE is included in your current Eclipse installation, do one of the following:
  - See list of installed plugins,
  - In Eclipse Oxygen go to **Window** > Perspective > **Open Perspective** > and choose **Remote System Explorer** or select Other and choose Remote System Explorer.
- Install Remote System Explorer plugin for Eclipse
  - Go to install new software to install the plugin
  - Under General Tools
    - Remote Services, Remote Command Shell Console, Remote system explorer end-user runtime

Work on a remote project with Eclipse via SSH

- To create an SSH remote project from the RSE perspective in Eclipse:
  - Define a new connection by going to File > New - > Remote System Explorer > connection
  - choose SSH only from the Select Remote System Type screen in the New Connection dialog.
  - Enter the connection information and select Finish.
  - Connect to the new host. (Assumes SSH keys are already setup.)
  - Once connected, drill down into the host's *Sftp Files*, choose a folder and select **Create Remote Project** from the item's context menu. (Wait as the remote project is created.)

References

1. **Computer Networking: a top-down approach (7th edition), by** JF Kurose, KW Ross, 2016
2. https://www.slideshare.net/raedald/http-vs-https-32902338