

# COSC 360: Topic 1 Continued

# Chapter 2

**1** Internet  
Protocols

**2** Domain Name  
System

**3** Uniform Resource  
Locators

**4** Hypertext Transfer  
Protocol

**5** Web Browsers

**6** Web Servers

**7** Summary




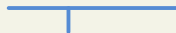
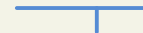
# URL Components



In order to allow clients to request particular resources from the server, a naming mechanism is required so that the client knows how to ask the server for the file.

For the web that naming mechanism is the **Uniform Resource Locator (URL)**.

`http://www.funwebdev.com/index.php?page=17#article`

				
Protocol	Domain	Path	Query String	Fragment

`ftp://example.com/abc.txt`

→ sends out an FTP request on port 21

`http://example.com/abc.txt`

→ transmits an HTTP request on port 80

# Uniform Resource Locators

## Domain

- The domain identifies the server from which we are requesting resources.
- Since the DNS system is case insensitive, this part of the URL is case insensitive.
- Alternatively, an IP address can be used for the domain:
  - <https://1.1.1.1>

# Uniform Resource Locators

## Port

- The optional port attribute allows us to specify connections to ports other than the defaults
- Add a colon after the domain, then specify an integer port number.

# Uniform Resource Locators

## Path

Familiar concept to anyone who has ever used a computer file system.

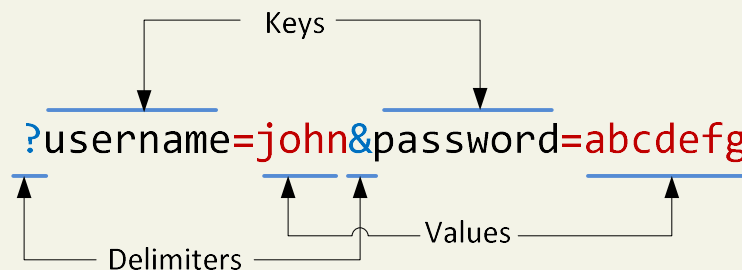
- The root of a web server corresponds to a folder somewhere on that server.
  - On many Linux servers that path is `/var/www/html/`
  - On Windows IIS machines it is often `/inetpub/wwwroot/`
- The path is optional
  - However, when requesting a folder or the top-level page, the web server will decide which file to send you.

# Query String



Query strings will be covered in depth when we learn more about HTML forms and server-side programming.

They are the way of passing information such as user form input from the client to the server. In URL's they are encoded as key-value pairs delimited by “&” symbols and preceded by the “?” symbol.



# Uniform Resource Locators

## Fragment

A way of requesting a portion of a page.

- Browsers will see the fragment in the URL, seek out the tag anchor in the HTML, and scroll the website to it.
- For example:
  - <https://funwebdev.com/some-page#section1>
- This would take us to the specified “section1” part of the “some-page” that we requested



# Chapter 2

**1** Internet  
Protocols

**2** Domain Name  
System

**3** Uniform Resource  
Locators

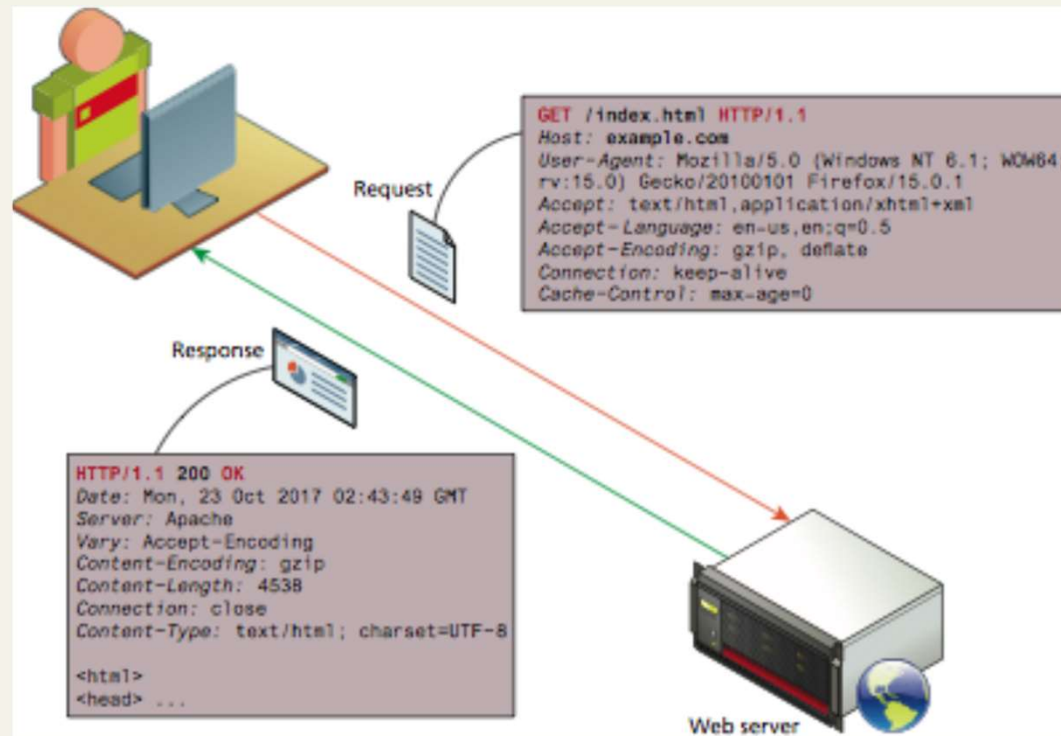
**4** Hypertext Transfer  
Protocol

**5** Web Browsers

**6** Web Servers

**7** Summary

# HTTP



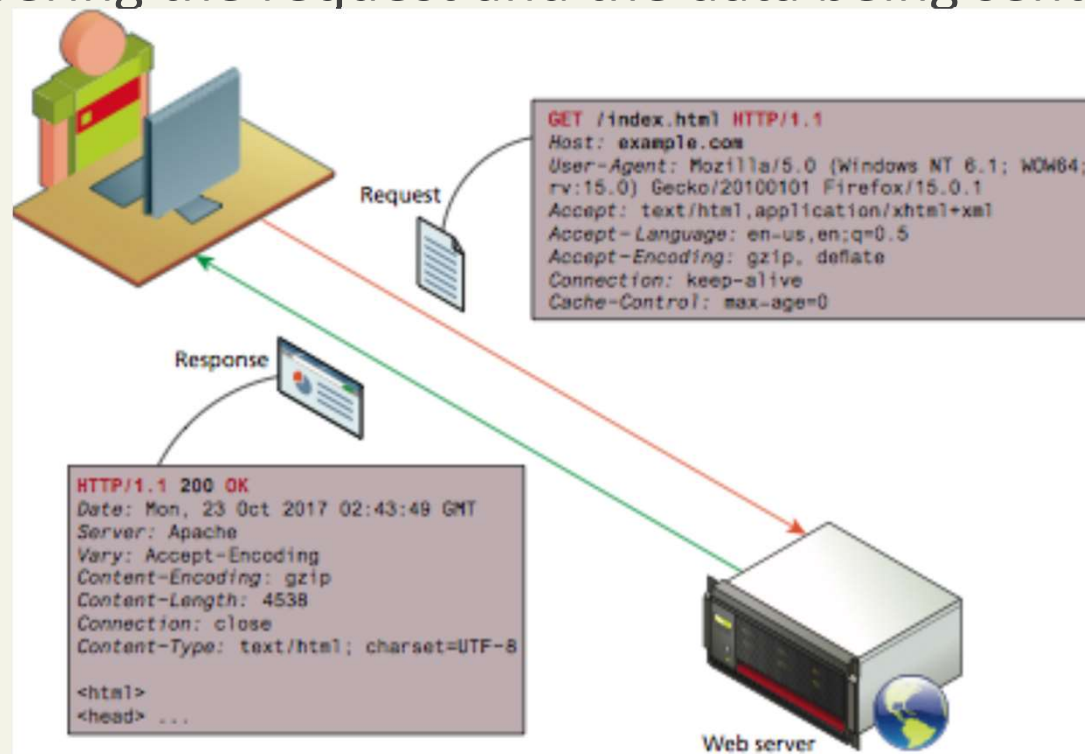
HTTP protocol: TCP connection on port 80 (by default).

Server waits for the request, and then responds with a response code, headers and an optional message (which can include files)

# Hypertext Transfer Protocol

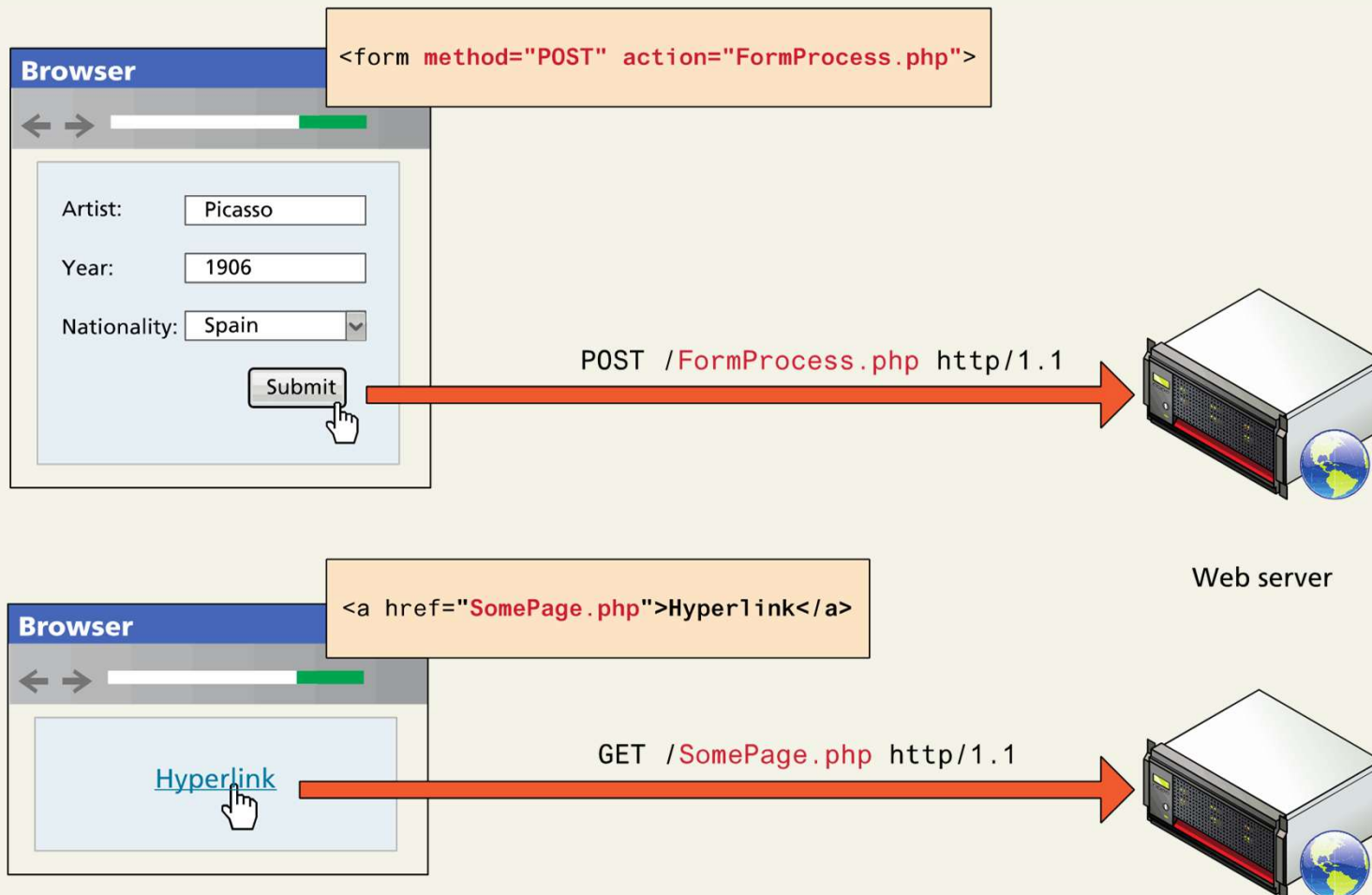
## Headers

- **Request headers** include data about the client machine.
- **Response headers** have information about the server answering the request and the data being sent



# Hypertext Transfer Protocol

## Request Methods



# Hypertext Transfer Protocol

## Response Codes

- 2## codes are for successful responses,
- 3## are for redirection-related responses,
- 4## codes are **client** errors,
- 5## codes are **server** errors.

200: OK

301: Moved Permanently

304: Not Modified

307: Temporary redirect

400: Bad Request

401: Unauthorized

404: Not found

414: Request URI too long

500: Internal server error

# Chapter 2

**1** Internet  
Protocols

**2** Domain Name  
System

**3** Uniform Resource  
Locators

**4** Hypertext Transfer  
Protocol

**5** Web Browsers

**6** Web Servers

**7** Summary

# Web Requests

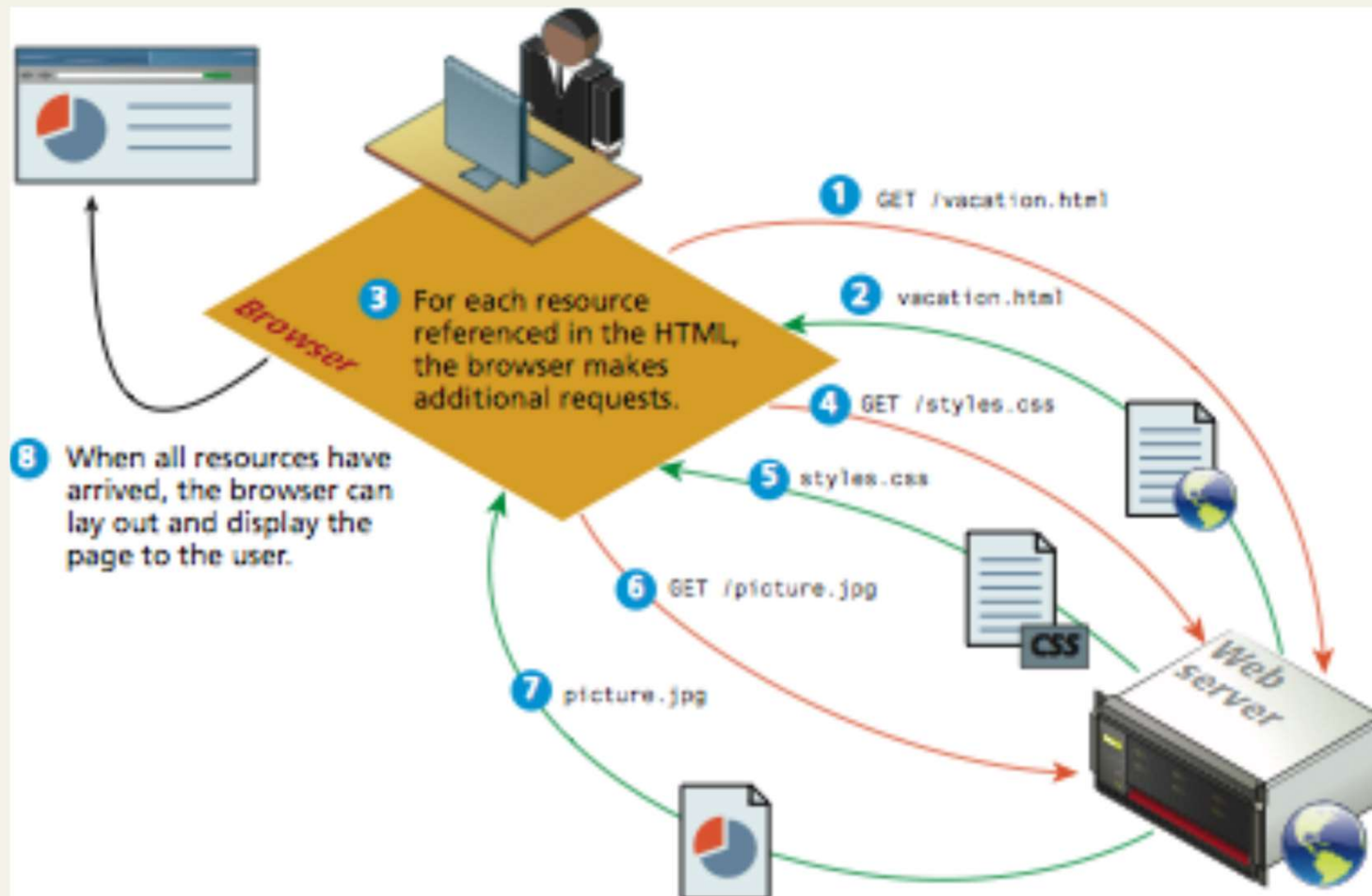


While we as web users might be tempted to think of an entire page being returned in a single HTTP response, this is not in fact what happens.

In reality the experience of seeing a single web page is facilitated by the client's browser which requests the initial HTML page, then parses the returned HTML to find all the resources referenced from within it, like images, style sheets and scripts.

Only when all the files have been retrieved is the page fully loaded for the user

# Browser parsing HTML and making subsequent requests





# Web Browsers

## Fetching a Web Page – Load Times and Cascades

The screenshot shows a web browser window with the address bar displaying `funwebdev.com/samples/chapters/chapter-3/`. The page title is "Chapter 3 | Fundamentals ...". The main content area features a large orange banner with the text "Introduction to CSS 3" and "CHAPTER OBJECTIVES". Below the banner, it lists the chapter objectives: "In this chapter you will learn ...", "The rationale for CSS", and "The syntax of CSS".

The browser's developer tools are open, showing the Network tab. The network log displays 75 requests, with a total size of 3,351.72 KB and a total time of 3.83 s. The log includes columns for Status, Method, File, Domain, Cause, Type, Transferred, Size, and timing information. The requests are as follows:

Status	Method	File	Domain	Cause	Type	Transferred	Size	0 ms	640 ms	1.28 s	1.92 s	2.56 s	3.20 s
304	GET	thumbs_chapter1-29.png	funwebdev.com	img	png	8.95 KB					142 ms		
304	GET	chapter3-95.png	funwebdev.com	imageset	png	47.66 KB					129 ms		
304	GET	slide-javascript-50x50.jpg	funwebdev.com	imageset	jpeg	2.12 KB					138 ms		
304	GET	dog-adoption-50x50.jpg	funwebdev.com	imageset	jpeg	2.09 KB					139 ms		
304	GET	Dollarphotoclub_92872465-web-50x50.jpg	funwebdev.com	imageset	jpeg	1.88 KB					134 ms		
304	GET	exam-takers-50x50.jpg	funwebdev.com	imageset	jpeg	2.39 KB					121 ms		
304	GET	adoptions2015-50x50.jpg	funwebdev.com	imageset	jpeg	2.08 KB					152 ms		
200	GET	like.php?href=http://funwebdev.com/samples/chapters/cha...	www.facebook.com	subdocument	html	8.40 KB	26.91 KB				114 ms		
200	GET	i-u7L8opqbVjs	www.facebook.com	xhr	js	389.56 KB							
304	GET	analytics.js	www.google-analytics.com	script	js	11.32 KB	27.15 KB				24 ms		
200	GET	collect?v=1&_v=j47&a=101546702&t=pageview&_s=1&...	www.google-analytics.com	img	gif	35 B	35 B				24 ms		
200	GET	like.php?href=http://funwebdev.com/samples/chapters/cha...	www.facebook.com	subdocument	html	8.37 KB	26.87 KB						
200	GET	i-u7L8opqbVjs	www.facebook.com	xhr	js	389.56 KB							
200	GET	style.css?ver=4.8.1	funwebdev.com	stylesheet	css	273.26 KB							
200	GET	rgs.css?ver=4.8.1	funwebdev.com	stylesheet	css	2.29 KB							

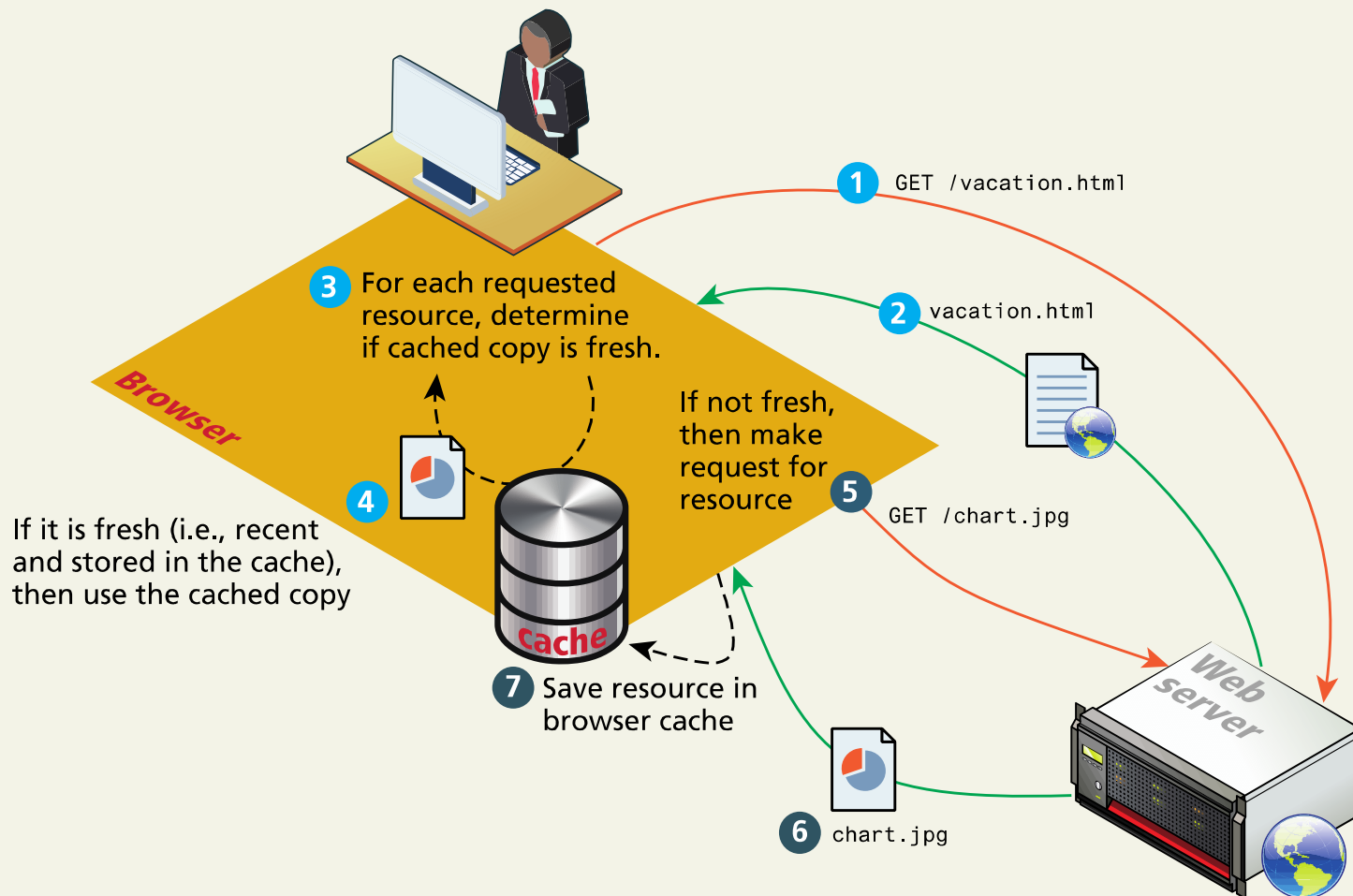
# Web Browsers

## Browser Rendering

- Interpreting the entire HTML markup together with the image and other assets into a grid of pixels for display within the browser window is called rendering the webpage.
- Implemented differently for each browser (Firefox, Chrome, Safari, Explorer, and Opera)

# Web Browsers

## Browser Caching



# Question

We want to fetch a new page from the web server if it isn't cached using cache control. What kind of request should we use, and where does the cache-control parameter go?

- A. HEAD, Request Body
- B. GET, Request Body
- C. GET, Request Headers
- D. POST, Request Headers

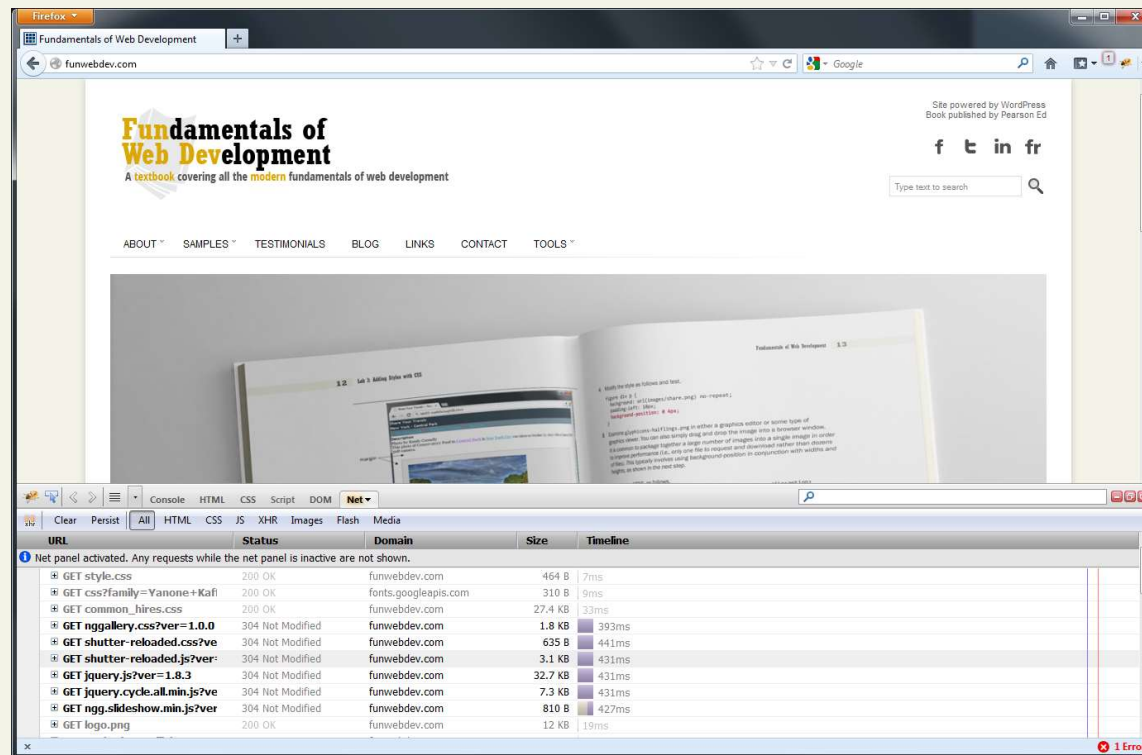
# Question

We want to fetch a new page from the web server if it isn't cached using cache control. What kind of request should we use, and where does the cache-control parameter go?

- A. HEAD, Request Body
- B. GET, Request Body
- C. GET, Request Headers**
- D. POST, Request Headers

# Browser Tools for HTTP

Modern browsers provide the developer with tools that can help us understand the HTTP traffic for a given page.



# Chapter 2

**1** Internet  
Protocols

**2** Domain Name  
System

**3** Uniform Resource  
Locators

**4** Hypertext Transfer  
Protocol

**5** Web Browsers

**6** Web Servers

**7** Summary

# Web Servers

A **web server** is, at a fundamental level, nothing more than a computer that responds to HTTP requests.

Real-world web servers are often more powerful than your own desktop computer

Webservers must choose an **application stack** to run a website. This application stack will include an

- operating system,
- web server software,
- a database,
- and a scripting language for dynamic requests

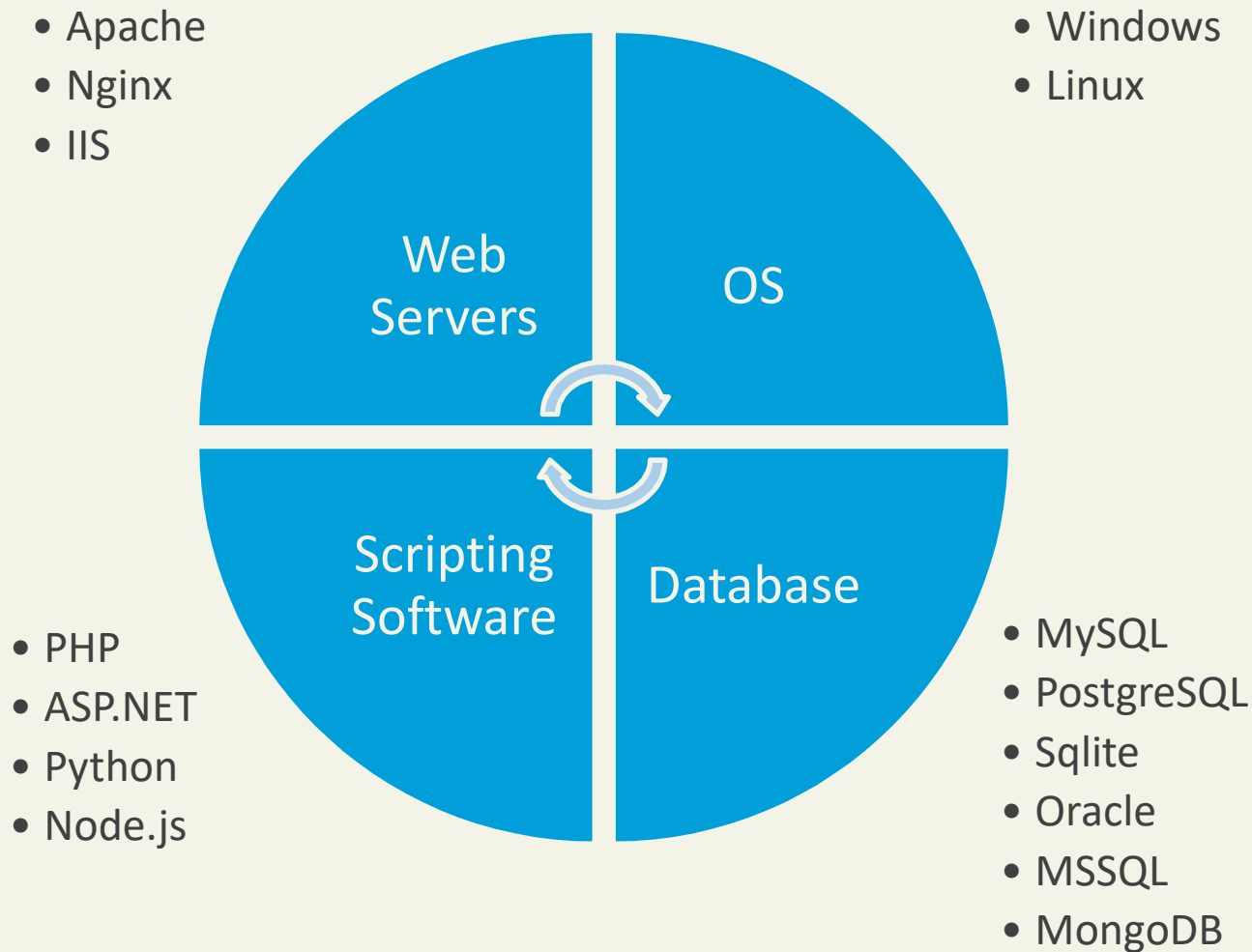


# LAMP Software Stack

Throughout this textbook we will rely on the **LAMP software stack**, which refers to the

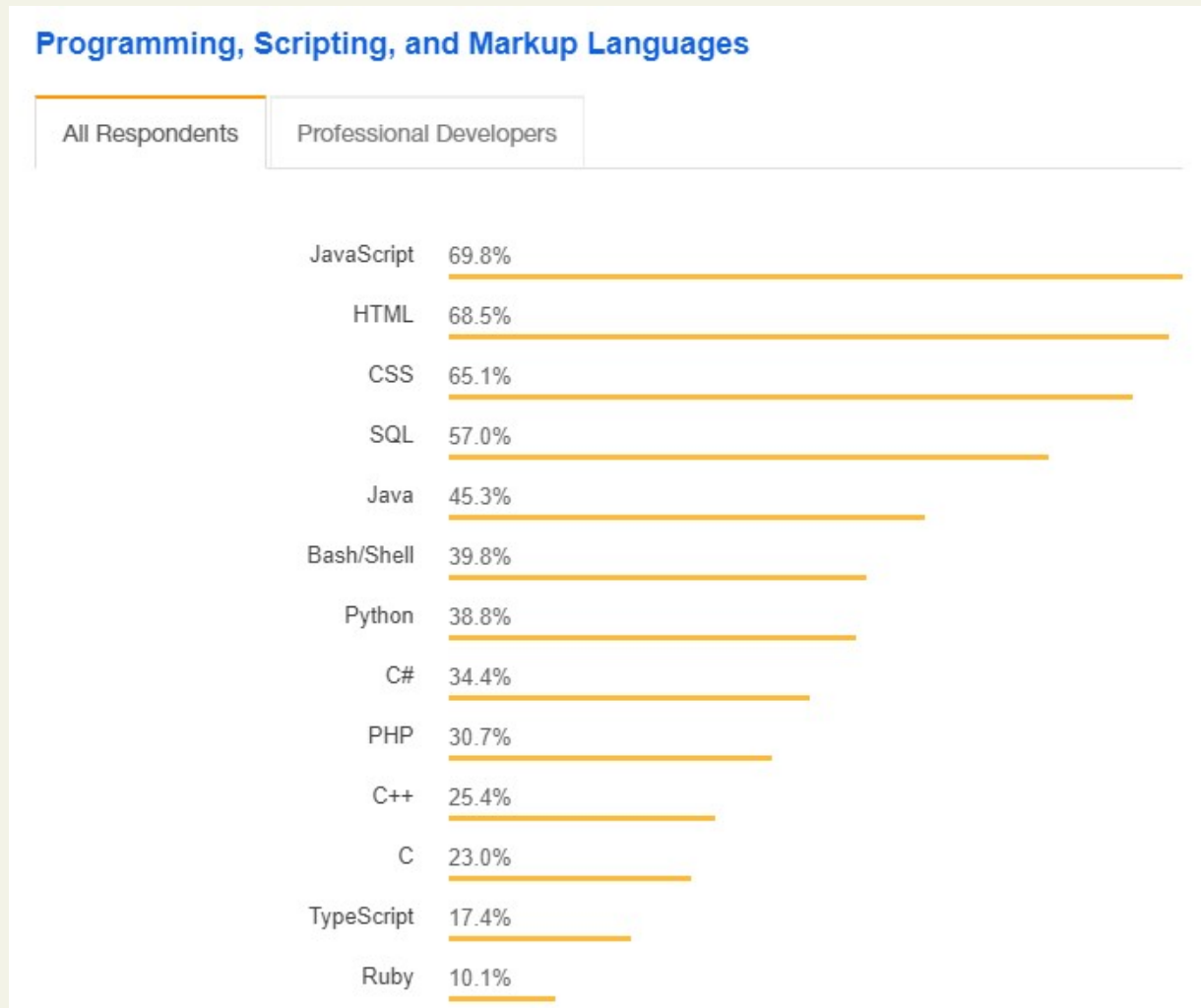
- Linux operating system
- Apache web server
- MySQL database,
- PHP scripting language

# Web Server Technologies



# Web Server Technologies

What are we using in 2018?



# Chapter 2

**1** Internet  
Protocols

**2** Domain Name  
System

**3** Uniform Resource  
Locators

**4** Hypertext Transfer  
Protocol

**5** Web Browsers

**6** Web Servers

**7** Summary

# Next Topic:

Topic 2: HTML Basics