How the Web Works

Chapter 1

Modified by Marissa Schmidt
Randy Connolly and Ricardo Hoar
Objectives

1. Definitions and History

2. Client-Server Model

3. Domain Name System

4. Uniform Resource Locators (URL)

5. Hypertext Transfer Protocol (HTTP)

6. Web Servers
DEFINITIONS AND HISTORY
Internet = Web?

The answer is no

The **World-Wide Web** (WWW or simply the Web) is certainly what most people think of when they see the word “internet.”

But the WWW is only a subset of the Internet.
Tim Berners-Lee

I meant Sir Tim Berners-Lee

The invention of the WWW is usually attributed to the British Tim Berners-Lee, who, along with the Belgian Robert Cailliau, published a proposal in 1990 for a hypertext system while both were working at CERN in Switzerland.
Core Features of the Web

Shortly after that initial proposal Berners-Lee developed the main features of the web:

1. A **URL** to uniquely identify a resource on the WWW.

2. The **HTTP protocol** to describe how requests and responses operate.

3. A software program (later called **web server software**) that can respond to HTTP requests.

4. **HTML** to publish documents.

5. A program (later called a **browser**) to make HTTP requests from URLs and that can display the HTML it receives.
Web Server Basics
Static Web Sites

Partying Like It’s 1995

In the early days, the skills needed to create a web site were pretty basic: one needed knowledge of the HTML markup language and perhaps familiarity with editing and creating images.

This type of web site is commonly referred to as a static web site, in that it consists only of HTML pages that look identical for all users at all times.
Static Web Sites

1. I want to see vacation.html

2. Server retrieves files from its hard drive

3. Server "sends" HTML and then later the image to browser

4. Browser displays files
Dynamic Web Sites

Within a few years of the invention of the web, sites began to get more complicated as more and more sites began to use programs running on web servers to generate content dynamically.
Dynamic Web Sites

1. I want to see vacation.php

2. Server recognizes that it must run a dynamic script that is on its hard drive.

3. Server executes or interprets the script.

4. Scripts "outputs" HTML

5. Server "sends" generated HTML and the image file to user.

6. Browser displays files
CLIENT-SERVER MODEL
Client-Server Model

What is it?

The web is sometimes referred to as a client-server model of communications.

In the **client-server model**, there are two types of actors: clients and servers.

The **server** is a computer agent that is normally active 24 hours a day, 7 days a week (or simply 24/7), listening for queries from any client who make a request.

A **client** is a computer agent that makes requests and receives responses from the server, in the form of response codes, images, text files, and other data.
Request-Response Loop

Within the client-server model, the request-response loop is the most basic mechanism on the server for receiving requests and transmitting data in response.

The client initiates a request to a server and gets a response that could include some resource like an HTML file, an image or some other data.
Section 5 of 8

DOMAIN NAME SYSTEM (DNS)
TCP/IP

Communication protocol multiple programs and services to share the same physical computer and internet connection using **ports**.

<table>
<thead>
<tr>
<th>TCP Port</th>
<th>Service</th>
</tr>
</thead>
<tbody>
<tr>
<td>21</td>
<td>file transfer (FTP)</td>
</tr>
<tr>
<td>22</td>
<td>secure shell (SSH)</td>
</tr>
<tr>
<td>25, 110</td>
<td>email (SMTP, POP3)</td>
</tr>
<tr>
<td>80</td>
<td>web (HTTP)</td>
</tr>
<tr>
<td>443</td>
<td>secure web (HTTPS)</td>
</tr>
</tbody>
</table>
Internet Protocol (IP)

The Internet uses the **Internet Protocol (IP)** addresses to identify destinations on the Internet.

Every device connected to the Internet has an **IP address**, which is a numeric code that is meant to uniquely identify it.
IP Addresses

Two types

IPv4
2^{32} addresses

4 - 8 bit components
(32 bits)

192.168.123.254

IPv6
2^{128} addresses

8 - 16 bit components
(128 bits)

Domain Name System

Why do we need it?

As elegant as IP addresses may be, human beings do not enjoy having to recall long strings of numbers. Instead of IP addresses, we use the Domain Name System (DNS)
DNS Overview

1. What’s the IP address of www.funwebdev.com?

2. Here it is, it’s: 66.147.244.79

3. I want the default page at 66.147.244.79

4. Here it is...
Domain Levels

Third-Level Domain

Top Level Domain (TLD)

server1.www.funwebdev.com

Fourth-Level Domain

Second-Level Domain (SLD)

Most general

Top-Level Domain (TLD)

Most specific

Second-Level Domain (SLD)

Third-Level Domain

Fourth-Level Domain

server1

com

funwebdev

www
UNIFORM RESOURCE LOCATORS (URL)
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
```

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Domain</th>
<th>Path</th>
<th>Query String</th>
<th>Fragment</th>
</tr>
</thead>
</table>

Randy Connolly and Ricardo Hoar
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.
HYPERTEXT TRANSFER PROTOCOL (HTTP)
HTTP

The HTTP protocol establishes a TCP connection on port 80 (by default).

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

GET /index.html HTTP/1.1
Host: example.com
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:15.0) Gecko/20100101 Firefox/15.0.1
Accept: text/html,application/xhtml+xml
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Cache-Control: max-age=0

HTTP/1.1 200 OK
Date: Mon, 22 Oct 2012 02:43:49 GMT
Server: Apache
Vary: Accept-Encoding
Content-Encoding: gzip
Content-Length: 4538
Connection: close
Content-Type: text/html; charset=UTF-8

<html>
<head> ...

Web server
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

1. GET /vacation.html
2. vacation.html
3. For each resource referenced in the HTML, the browser makes additional requests.
4. GET /styles.css
5. styles.css
6. GET /picture.jpg
7. picture.jpg
8. When all resources have arrived, the browser can lay out and display the page to the user.
HTTP Request Methods

The HTTP protocol defines several different types of requests, each with a different intent and characteristics.

The most common requests are

- **GET** - Requests a specific file/resource from the server.
- **POST** - Submits form information to the server.
- **PUT** - Uploads a file to the server.
- **HEAD** - Requests information about a file from the server, but not the file’s contents.

Other requests, such as DELETE, CONNECT, TRACE and OPTIONS are seldom used, and are not covered here.
GET versus POST requests

<form method="POST" action="FormProcess.php">
  Artist: Picasso
  Year: 1906
  Nationality: Spain
  Submit

POST /FormProcess.php http/1.1

<a href="SomePage.php">Hyperlink</a>

GET /SomePage.php http/1.1
Browser Tools for HTTP

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

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

Regardless of the physical characteristics of the server, one must choose an application stack to run a website.

This stack will include an operating system, web server software, a database and a scripting language to process 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, and **PHP** scripting language.
Many corporations, for instance, make use of the Microsoft WISA software stack, which refers to Windows operating system, IIS web server, SQL Server database, and the ASP.NET server-side development technologies.
What You’ve Learned

1. Definitions and History
2. Client-Server Model
3. Domain Name System
4. Uniform Resource Locators (URL)
5. Hypertext Transfer Protocol (HTTP)
6. Web Servers