

### 1.2.3 Internet principles of operation

#### WORKING OF THE INTERNET

Whenever you need to send a message or retrieve information from another computer, the TCP/IP protocols are what make the transmission possible. Your request goes out over the network, hitting **domain name servers (DNS)** along the way to find the target server. The DNS points the request in the right direction.

Once the target server receives the request, it can send a response back to your computer. The data might travel a completely different path to get back to you. This flexible approach to data transfer is part of what makes the Internet such a powerful tool.

#### **BROWSER:**

Browser, short for *web browser*, is a software application used to enable computers users to locate and access web pages. Browsers translates the basic HTML (Hypertext Mark Up Language) code that allows us to see images, text videos and listen to audios on websites, along with hyperlinks that let us travel to different web pages. The browser gets in contact with the web server and requests for information. The web server receives the information and displays it on the computer.

There are several types of browsers but the one thing they have in common is that they carry out the same function; they transfer hypertext. Here are some familiar common web browsers:

- Google Chrome

- Mozilla Firefox

- Microsoft Internet Explorer

Every browser features a toolbar that allows you to perform various functions like: Go back to the first page you started on the internet which is called Home.

- Book your favorite websites

- Print content you find interesting on web pages

- Check your web history, like the websites you visited in the past

- You can go forward and backwards to see the previous sites you viewed

### 1.2.3 Internet principles of operation

There are two basic ways that you can navigate through the browser:

You can click on links on different web pages. Those links will lead you to other web pages.

If you know the URL or the address of the website you can type it in write in the browser's box all the way at the top. There's no need to type in the: `http://` part when inserting the address because the browser automatically places it in. Then you have to click enter.

So every time you click on a link or enter a URL, your browser will request the information from the web server. If your browser returns to the page with a error message, the address you

typed in could be wrong on you may not be connected to the internet.

### **What are Internet Servers?**

In general, all of the machines on the Internet can be categorized as two types: servers and clients. Those machines that provide services (like Web servers or FTP servers) to other machines are **servers**. And the machines that are used to connect to those services are **clients**. When you connect to “Yahoo!” at www.yahoo.com to read a page, Yahoo! is providing a machine (probably a cluster of very large machines), for use on the Internet, to service your request. Yahoo! is providing a server. Your machine, on the other hand, is probably providing no services to anyone else on the Internet. Therefore, it is a user machine, also known as a client. It is possible and common for a machine to be both a server and a client, but for our purposes here you can think of most machines as one or the other.

A server machine may provide one or more services on the Internet. For example, a server machine might have software running on it that allows it to act as a Web server, an e-mail server and an FTP server. Clients that come to a server machine do so with a specific intent, so clients direct their requests to a specific software server running on the overall server machine. For example, if you are running a Web browser on your machine, it will most likely want to talk to the Web server on the server machine. Your Telnet application will want to talk to the Telnet server, your e-mail application will talk to the e-mail server, and so on.

### **Hypertext Transfer Protocol (http)**

Short for Hypertext Transfer Protocol, the underlying protocol used by the World. HTTP defines how messages are formatted and transmitted, and what actions Web servers and browsers should take in response to various commands. For example, when you enter a URL in your browser, this actually sends an HTTP

command to the Web server directing it to fetch and transmit the requested Web page. There is much more to HTTP than simply sending the message that an end user wants to see a web page. HTTP actually identifies eight different methods which indicate what the desired action is and what action should be performed. These eight different methods are:

**HEAD** – Asks for the material in a GET request, but without the response body.

**GET** – Requests the representation of a specified source with a response body.

**POST** – Submits the data to be processed from an HTML form to the identified source.

**PUT** – Uploads a representation of the source that has been specified.

**DELETE** – Deletes a specific source.

**TRACE** – This echoes back a specific request so that the user can see what servers are adding or changing the request.

**OPTIONS** – Used to check the functionality of a web server by returning the HTTP methods that the server supports for a specified URL.

**CONNECT** – Facilitates SSL-encrypted communication through an unencrypted HTTP proxy by converting the request connection to a transparent TCP/IP tunnel.

Important to communication and the way the entire Internet works, the hypertext transfer protocol or HTTP is a necessary protocol.

The other main standard that controls how the World Wide Web works is HTML, which covers how Web pages are formatted and displayed.

## **HTML**

HTML is an abbreviation that stands for Hypertext Markup Language. It is the language of web pages—

the markup language that browsers read to render web pages.

HTML refers to a markup language that has several versions and web browsers and other user agents have varying levels of support for these versions. In general, the earlier a version is, the more browser support there is for that version.

## **HTML BASICS**

To publish information for global distribution, one needs a universally understood language, a kind of publishing mother tongue that all computers may potentially understand. The publishing language used by the World Wide Web is HTML (from Hypertext Markup Language).

HTML gives authors the means to:

Publish online documents with headings, text, tables, lists, photos, etc.

Retrieve online information via hypertext links, at the click of a button.

Design forms for conducting transactions with remote services, for use in searching for information, making reservations, ordering products, etc.

Include spread-sheets, video clips, sound clips, and other applications directly in their documents.

## **HTML Examples**

HTML is made up of elements with attributes, some of the most common ones you would see include:

`<p>` for paragraphs

`<a>` for links

`<div>` for dividing up sections of a page

## **HTML Versions**

There are a number of versions of HTML that have been supported by web browsers:

HTML 2.0

this was published as an IETF RFC in 1995. It was supported by some browsers like Mosaic.

HTML 3.2

this version was the first W3C recommendation. It had wider browser support (partially because there were a lot more browsers). It became a recommendation in January 1997.

HTML 4.0 and HTML 4.01

In December 1997, the W3C released this upgrade to HTML 3.2 as a recommendation. It added more features and tags and provided three versions: transitional, strict, and frameset. This was updated to 4.01 in December 1999 with a few changes to the specification.

## XHTML 1.0

XHTML 1.0 is a reformulation of HTML 4.01 under XML rules, and it was published as a recommendation by the W3C in January 2000. It has much stricter syntax and requires that any XHTML be valid and well-formed in order to display correctly. Most web browsers render XHTML

1.0 documents the same as they render HTML 4.01 documents.

## HTML5

HTML5 began being developed in 2004, when the W3C HTML working group decided to merge the HTML and XHTML tree to make HTML a purely XML-based language. This left designers and browser manufacturers who wanted a more flexible solution with the choice to give up or create their own new specification. They created a new group called the Web Hypertext Application Technology Working Group or WHATWG. HTML5 became a W3C working draft in 2008.

## **WEB PAGES:**

Web pages comprise of five basic layers:

1. Content
2. Semantics
3. Structure
4. Presentation
5. Behavior

Content refers to the text, images and multimedia components of web pages. *Behavior* refers mainly to JavaScript. Here we will discuss rest of the three layers.

### Semantics

With regards to theme development, semantics refers to use of markup (HTML) to convey meaning rather than presentation. For example, we can use list code (ol, ul, li, dl, dt, dd) to say "this is a list" or heading tags such as h1, h2, h3 to say "these are headings and subheadings."

### Structure

Structure refers to the practice of using semantic HTML to describe how blocks of information relate to one another.

The relations between data are primary built using proximity and correct use of semantic elements. You can think of proximity as simply content source order—that is, the order in which the content appears in the HTML. With the correct use of semantic HTML we can describe what content items *mean in terms of the content around it*.

## **Source Order and Linearised Content**

Source order is the order in which the content appears in the markup. Correctly linearised content makes sense when there is no layout applied to the page, such as viewing a text only version of the page, for example in the Lynx browser or some mobile phone browsers.

Tool: RightLynx browser add-on for Firefox and Internet Explorer (emulates Lynx).

Web page structure is vitally important to accessibility because the content must make sense when linearised.

Many web pages use tables for page layout. This is frowned upon because it's difficult to linearize the content. It is strongly advised to use CSS for layout rather than tables.

Please note that in this context we are just talking about the deprecated practice of using tables for layout design. Of course you should use tables to mark up tabular data (such as spreadsheets) and there are special considerations for building accessible tables.

By using CSS for layout, it is possible to structure the source order so that it makes logical sense when linearised, yet the visual layout can be different.

### **Structural Markup to Provide Meaning**

Primarily this pertains to the use of heading tags, such as <h1>, <h2> and <h3> to break pages into logical sections and subsections.

There are a few simple rules to follow when using headings:

1. Only one h1 tag is allowed per page.
2. Every page should have an h1 heading.
3. Headings must be properly nested.

### **Presentation**

HTML should never be used for presentation. Use of the font tag and many other HTML 4 elements are now deprecated in favor of their more semantic replacements. In order to style web page content you should use CSS, preferably in an external style sheet.

By using CSS to separate the content from the presentation you can improve the accessibility of web pages in a number of ways:

Provide meaning to your documents without worrying about what it looks like (CSS will take care of the style).

Structure the main content to be first in the markup, but display it anywhere you want (CSS will provide the layout).

## **MARKUP**

Markup comprises a set of symbols, or a language, that can be used to provide instructions. The use of markup is supported by rules that define the symbols and how they should be used: a *specification*.

A webpage is created by marking-up content (text, images, etc.) using the Hypertext Markup Language (HTML).

The markup provides instructions to the software used for viewing a webpage (web browser) on how the page should look and work.

### **Markup and typesetting**

Markup is not exclusive to web and computer-based technologies. Prior to electronic publishing, plain-text would be marked-up with instructions to the typesetter. These instructions would indicate that text were a heading, quotation, caption, etc. Based on the markup, the text would be *styled* using the appropriate typeface, point size, weight, etc.

### **HTML markup**

To create an HTML document, content is marked-up into information elements. Specific

tags, denoted by characters within angle brackets, are used to signify the beginning, and end of each element.

For example, to tell the web browser that ‘The Motive Web Design Glossary’ is a heading, the text is marked-up into a heading element, using level-one heading tags;

```
<h1>The Motive Web Design Glossary</h1>
```

The look of the heading, the typeface, type size and color, are set using a separate, supplementary presentation language: Cascading Style Sheets (CSS).

HTML elements include:

Headings,

Lists,

Tabulated data (tables),

Anchors,

Images

Quotations.

### **Presentational markup**

Although the HTML language specification defines the markup that can be used to create a webpage, the *webpage author* decides which elements they will use to markup content.

If a webpage author is using software that enables a webpage to be edited visually, rather than at a code level, they may be more inclined to use markup that achieves a visual effect—this is referred to *presentational markup*.

For example, the *appearance* of a heading may be created by making text larger, bold and a different color.

Presentational markup for such a heading may look like this:

```
<p><font size="+7" color="#990000"><b>The Motive Internet Glossary</b></font></p>
```

Similarly, the author could use either presentational or structural markup to display items as a list.

Approach to markup

Sample code

Appearance

Presentational

```
<p>
```

```
- eggs
```

```
- eggs<br />
```

```
- milk
```

```
- cheese
```

```
- milk<br />
```

```
- cheese
```

```
</p>
```

Structural

```
<ul>
```

```
□
```

```
eggs
```

```
<li>eggs</li>
□
<li>milk</li>
milk
<li>cheese</li>
□
cheese
</ul>
```

The structural markup example uses an unordered list element, `<ul></ul>` and list-item elements, `<li></li>`.

### **1.2.3 Internet principles of operation**

Limitations of presentational markup

As a general rule, presentational markup should be avoided;

Presentational markup is less efficient to maintain.

For example, if presentational markup is used, updating the visual design of a website involves editing the markup of every page which is quite tedious.

Presentational markup limits the ease with which content may be reinterpreted/repurposed.

For example, at a visual level, ‘large red text’ may communicate ‘heading’, but is just ‘more text’ to a computer program such as a search engine spider.

Support for presentational elements, such as the font element, has been phased out of contemporary markup specifications as more appropriate/efficient means of achieving the same results are introduced.

Future web browsers may not correctly display documents that use these superseded **(deprecated)** elements.

### **Structural markup**

#### **STRUCTURAL MARKUP IN WORD**

A non-web example of the benefits of a structural approach to content markup can be found when using Microsoft Word. If headings are used consistently throughout a document, then a table of contents can be generated automatically. If text has only been styled to look like a heading, then the program cannot distinguish headings from body copy, and the resulting table of contents is likely to be meaningless.

Structural markup is when elements are used to describe the structure of a document (webpage).

The HTML 4 language specification contains a limited number of structural elements, including: (meta) title;

Headings: levels 1-6;

Tables: tables can include captions, and markup that ‘binds’ column and row headings to data cells; Quotations: identifying content that has been referenced from another document;

Lists: unordered (bulleted), numbered, definition.

Notice that the list of available elements does not include any of the structural distinctions that are common to modern Webpages. For example, there are no elements for ‘masthead/banner’, ‘navigation menu’ or ‘related links’.

In place of the missing structural elements, webpage authors typically settle for headings.

## **Media Access Control (MAC) address:**

The MAC address is a unique value associated with a network adapter. MAC addresses are also known as hardware addresses or physical addresses. They uniquely identify an adapter on a LAN.

MAC addresses are 12-digit hexadecimal numbers (48 bits in length). By convention, MAC addresses are usually written in one of the following two formats:

MM:MM:MM:SS:SS:SS

MM-MM-MM-SS-SS-SS

The first half of a MAC address contains the ID number of the adapter manufacturer. These IDs are regulated by an Internet standards body. The second half of a MAC address represents the serial number assigned to the adapter by the manufacturer.

In the example “00:A0:C9:14:C8:29”

The prefix “00A0C9”

Indicates the manufacturer is Intel Corporation.

### **Why MAC Addresses?**

Recall that TCP/IP and other mainstream networking architectures generally adopt the OSI model. In this model, network functionality is subdivided into layers. MAC addresses function at the data link layer (layer 2

in the OSI model). They allow computers to uniquely identify themselves on a network at this relatively low level.

### **MAC vs. IP Addressing**

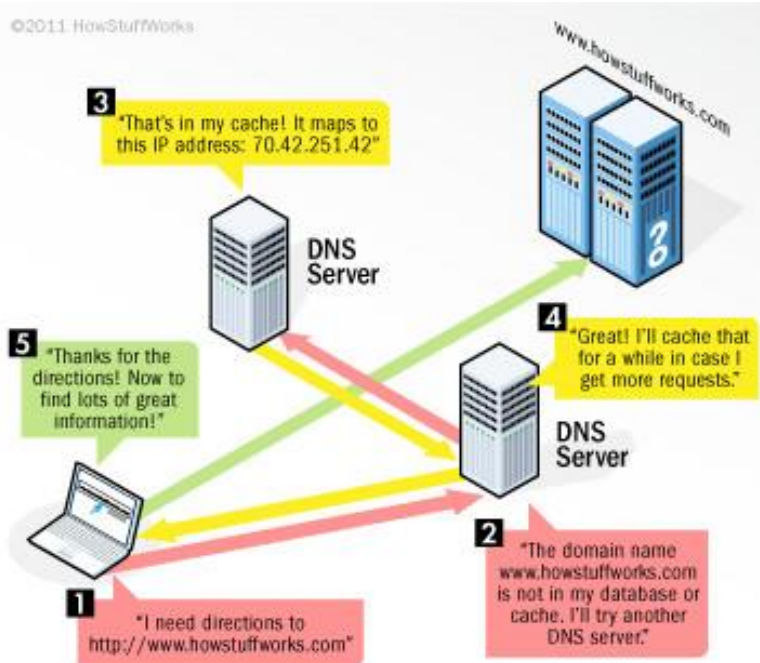
Whereas MAC addressing works at the data link layer, IP addressing functions at the network layer (layer 3).

It's a slight oversimplification, but one can think of IP addressing as supporting the software implementation and MAC addresses as supporting the hardware implementation of the network stack. The MAC address generally remains fixed and follows the network device, but the IP address changes as the network device moves from one network to another.

IP networks maintain a mapping between the IP address of a device and its MAC address. This mapping is known as the ARP cache or ARP table. ARP, the Address Resolution Protocol, supports the logic for obtaining this mapping and keeping the cache up to date.

DHCP also usually relies on MAC addresses to manage the unique assignment of IP addresses to devices.





### **Cookies:**

Cookies are very small text files placed on your computer by a Web server when you view some sites online (not all Web sites place cookies). They're used to store data about you and your preferences so that the Web server doesn't have to repeatedly request this information, potentially slowing down load time.

Cookies are commonly used to store personal registration data like your name, your address, the contents of a shopping cart, and your preferred layout for a Web page, what map you might be looking at, and so on.

Cookies make it easy for Web servers to personalize information to fit your specific needs and preferences when you're visiting a Web site.

### **Implications of Cookies:**

The easiest answer is that cookies, in and of themselves, are completely harmless. However, some Web sites use them to track users as they browse the Web, collecting highly personal information and often discreetly transferring that information to other Web sites without permission or warning. This is why we often hear about Web cookies in the news.

### **DNS:**

It's like your computer's GPS for the Internet. Its basic job is to turn a user-friendly domain name like "google.com" into an Internet Protocol (IP) address like 64.233.167.104 that computers use to identify each other on the network.

Computers and other network devices on the Internet use an IP address to route your request to the site you're trying to reach. This is similar to dialing a phone number to

connect to the person you're trying to call. Thanks to DNS, though, you don't have to keep your own address book of IP addresses. Instead, you just connect through a domain name server, also called a DNS server or name server, which manages a massive database that maps domain names to IP addresses.

## **Internet Protocol (IP) address**

Every machine on a network has a unique identifier. Just as you would address a letter to send in the mail, computers use the unique identifier to send data to specific computers on a network. Most networks today, including all computers on the Internet, use the TCP/IP protocol as the standard for communicating on the network. In the TCP/IP protocol, the unique identifier for a computer is called its IP address.

There are two standards for IP addresses: IP Version 4 (IPv4) and IP Version 6 (IPv6). All computers with IP

addresses have an IPv4 address, and many are starting to use the new IPv6 address system as well. Here's what these two address types mean:

IPv4 uses 32 binary bits to create a single unique address on the network. An IPv4 address is expressed by four numbers separated by dots. Each number is the decimal (base-10) representation for an eight-digit binary (base-2) number, also called an octet. For example:  
216.27.61.137

IPv6 uses 128 binary bits to create a single unique address on the network. An IPv6 address is expressed by eight groups of hexadecimal (base-16) numbers separated by colons, as in 2001:cdba:0000:0000:0000:3257:9652. Groups of numbers that contain all zeros are often omitted to save space, leaving a colon separator to mark the gap (as in 2001: cdba: 3257:9652).