Web Designing

Static Web Page

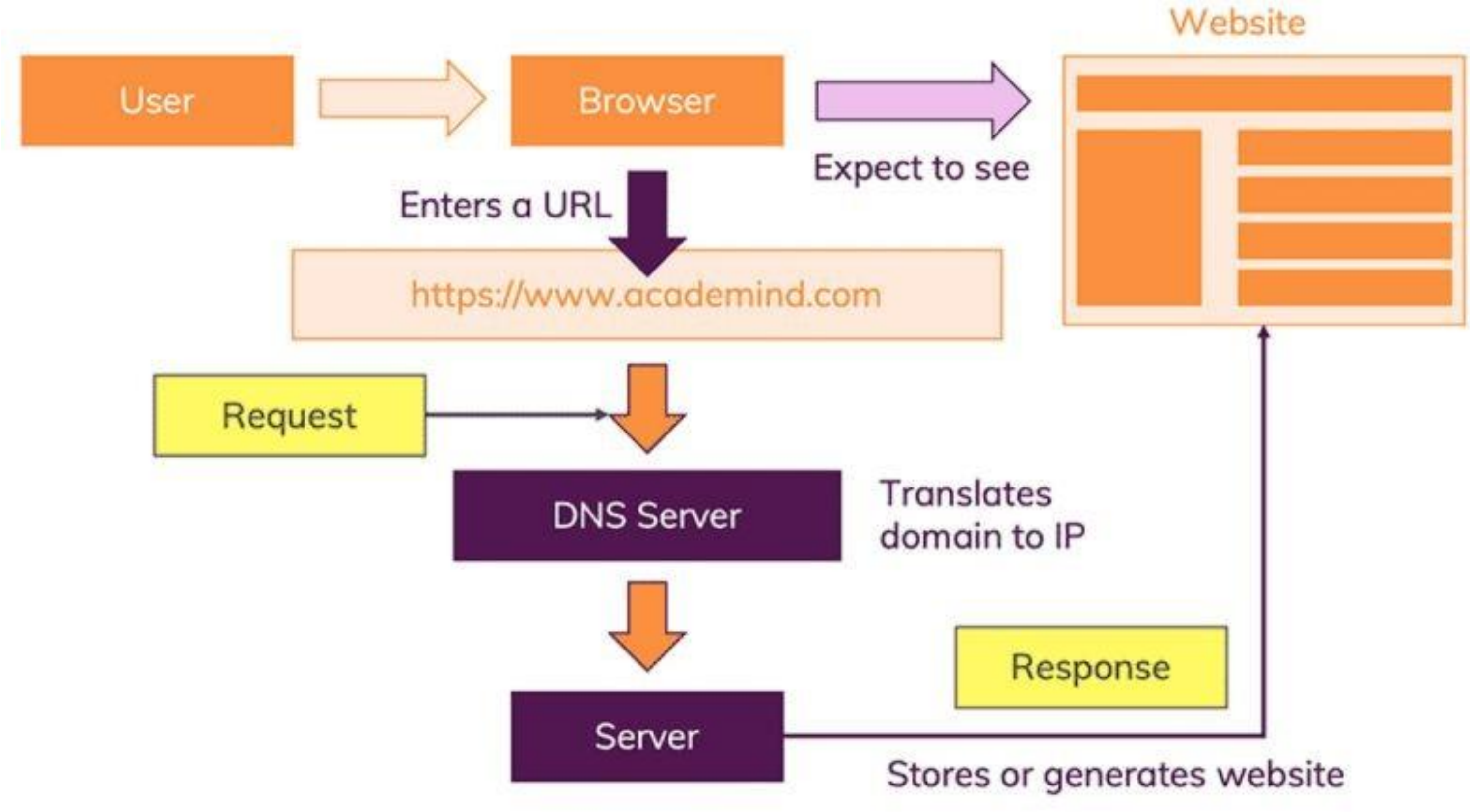
How Websites Work

Let's start with the most obvious way of using the internet: Visiting a website
svnit.ac.in

List of processes gets executed after hitting Enter.

- The URL gets resolved
- A Request is sent to the server of the website
- The response of the server is parsed
- The page is rendered and displayed

Overview Execution Flow



Step 1 - URL Gets Resolved

- The **website code** is obviously not stored on your machine and hence **needs to be fetched from another computer** where it is stored.
- This "other computer" is called a "**server**". Because it serves some purpose, in our case, it serves the website.
- "svnit.ac.in" is called "**a domain**" but actually, the server which hosts the source code of a website, is identified via **IP (Internet Protocol) addresses**.
- The browser sends a **"request"** to the server with the IP address you entered (indirectly - user entered "svnit.ac.in").

Domain

In reality, we often enter **"svnit.ac.in/cse"** like that. **"svnit.ac.in" is the domain**, **"/cse" is the path**.

Together, they make up the **"URL" ("Uniform Resource Locator").**

In addition, you can visit most websites via "www.svnit.ac.in" or just "svnit.ac.in".

Technically, "www" is a subdomain but most websites simply redirect traffic to "www" to the main page.

IP Address

An IP address typically looks like this: 172.56.180.5

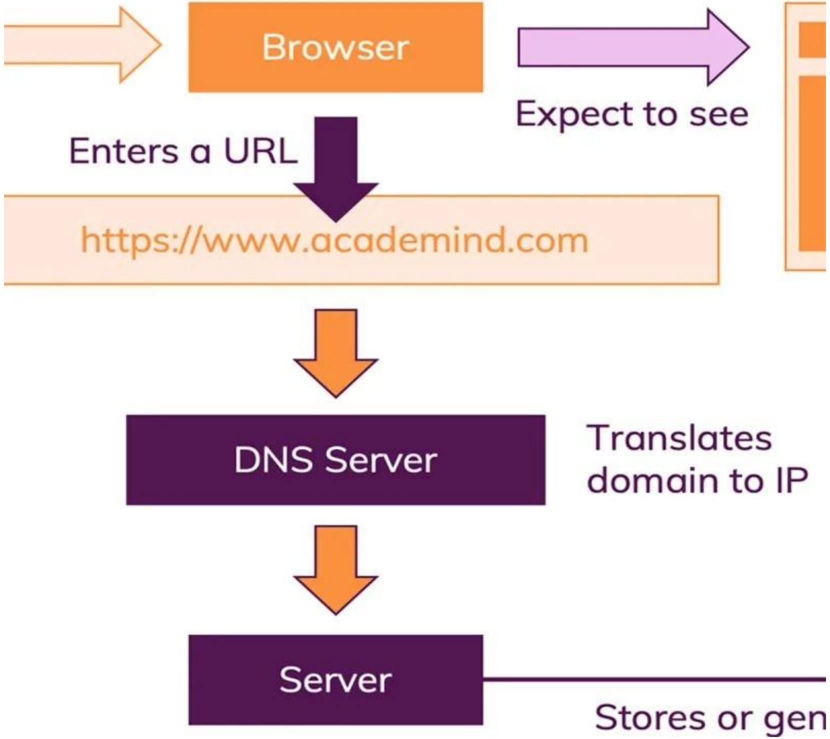
More "modern" form of IP Address is called IPv6.

How is the **domain "svnit.ac.in" translated to its IP address**?

- Special type of server in the internet called **"name server"** or **"DNS server"** (where DNS = "Domain Name System").
- DNS servers **translate domains to IP addresses**. You can imagine those servers as huge dictionaries that store translation tables: Domain => IP address.

When you enter **"svnit.ac.in"**, the **browser therefore first fetches the IP address from such a DNS server**.

DNS Server


Browser
Expect to see
Enters a URL
https://www.academind.com
DNS Server
Translates domain to IP
Server
Stores or gen

Step 2 - Request Is Sent

With the IP address resolved, the **browser** goes ahead and **makes a request to the server with that IP address**.

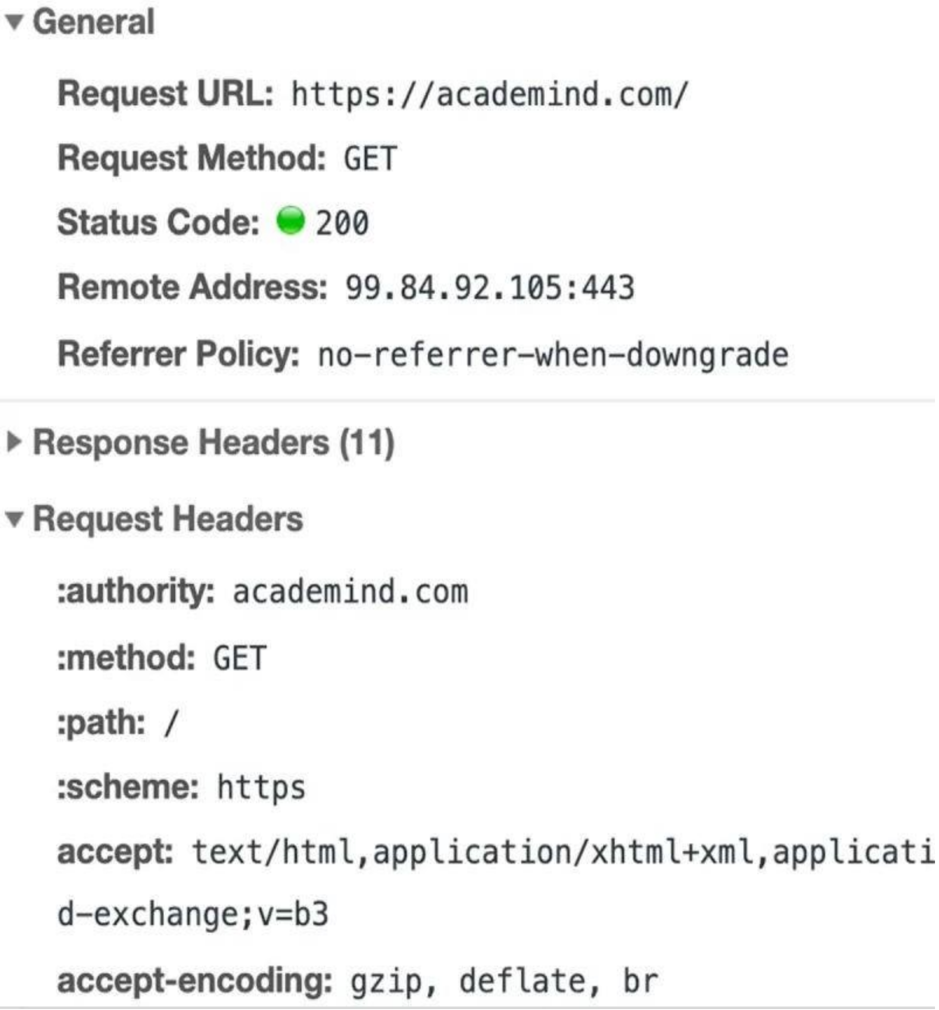
"A request" prepared by the browser. The browser bundles up a bunch of information that may contains:

- What's the exact URL?
- Which kind of request should be made?
- Should metadata be attached

and sends that data package to the IP address.

Request Example:

The data is sent via the **"HyperText Transfer Protocol" (known as "HTTP")** - a standardized protocol which defines what a request (and response) has to look like, which data may be included (and in which form) and how the request will be submitted.



▼ General

Request URL: https://academind.com/

Request Method: GET

Status Code: ● 200

Remote Address: 99.84.92.105:443

Referrer Policy: no-referrer-when-downgrade

▶ Response Headers (11)

▼ Request Headers

:authority: academind.com

:method: GET

:path: /

:scheme: https

accept: text/html,application/xhtml+xml,applicati

d-exchange;v=b3

accept-encoding: gzip, deflate, br

Request

A full URL actually looks like this: http://svnit.ac.in.com. The browser **auto-completes** it for you.

And there also is HTTPS - it's like HTTP but encrypted. Most modern pages use that instead of HTTP. A full URL then becomes: https://svnit.ac.in

Since the whole process and format is standardized, there is no guessing about how that request has to be read by the server.

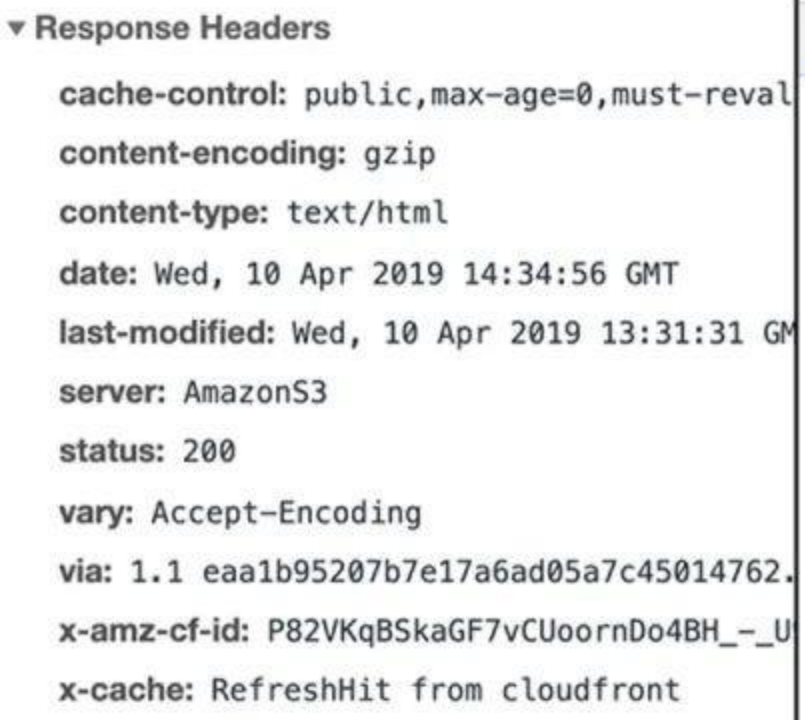
Response

The server handles the request appropriately and returns a "response".

A "response" is a technical thing and kind of similar to a "request". We could say it's basically a "request" in the opposite direction.

Like a request, **a response can contain data, metadata etc**. When requesting a page like svnit.ac.in, **the response will contain the code that is required to render the page onto the screen**.

Example



▾ Response Headers

cache-control: public,max-age=0,must-reval

content-encoding: gzip

content-type: text/html

date: Wed, 10 Apr 2019 14:34:56 GMT

last-modified: Wed, 10 Apr 2019 13:31:31 GM

server: AmazonS3

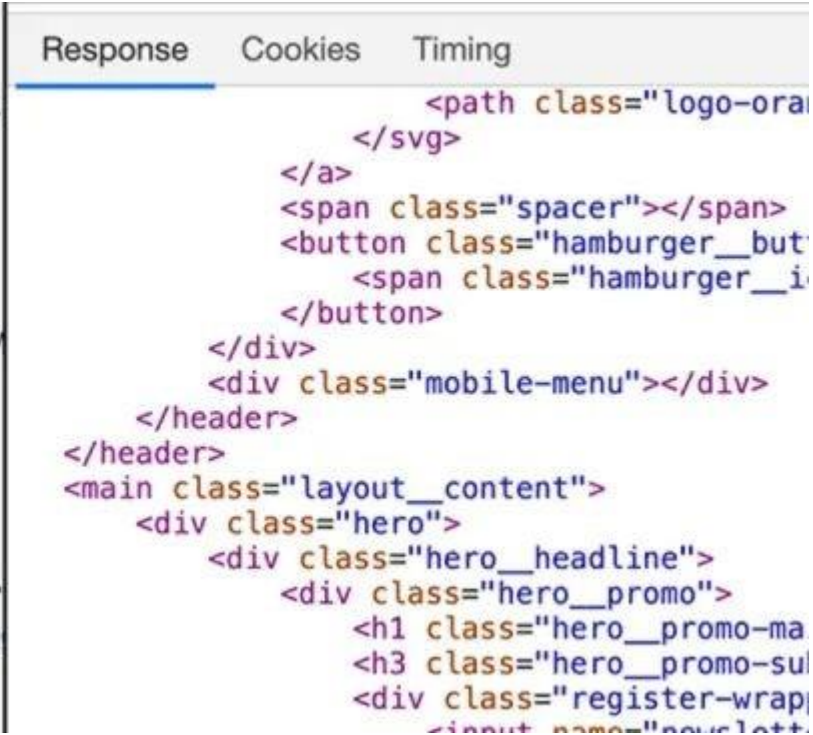
status: 200

vary: Accept-Encoding

via: 1.1 eaa1b95207b7e17a6ad05a7c45014762.

x-amz-cf-id: P82VKqBSkaGF7vCUoornDo4BH_-_U

x-cache: RefreshHit from cloudfront



```
          <path class="logo-ora
        </svg>
      </a>
      <span class="spacer"></span>
      <button class="hamburger__but
        <span class="hamburger__i
      </button>
    </div>
    <div class="mobile-menu"></div>
  </header>
</header>
<main class="layout__content">
  <div class="hero">
    <div class="hero__headline">
      <div class="hero__promo">
        <h1 class="hero__promo-ma
        <h3 class="hero__promo-su
        <div class="register-wrap
          <input name="newslett
```

What happens on the server?

That's defined by **web developers**.

In the end, a response has to be sent. That response doesn't have to contain "a website". It can **contain any data - including files or images**.

Some servers are programmed to generate websites dynamically based on the request (e.g. a profile page that contains your personal data), other servers return pre-generated HTML pages (e.g. a news page). Or both is done - for different parts of a webpage.

There also is a third alternative: Websites that are pre-generated but that change their appearance and data in the browser.

Step 3 - Response Is Parsed

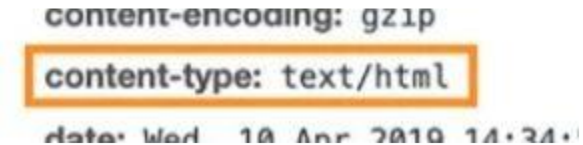
The **browser receives the response sent by the server**. This alone, doesn't display anything on the screen though.

Instead, the next step is that the **browser parses the response**.

The browser checks the data and metadata that's enclosed in the response. And based on that, it decides what to do.

You might've had cases where a PDF opened in your browser. That happened because the response informed the browser that the data is not a website but a PDF document instead. And the browser tries to pick the best handling mechanism for any data type it detects.

In case of website, the response would contain a specific piece of metadata, that tells the browser that the response data is of type text/html.



```
content-encoding: gzip
content-type: text/html
date: Wed, 10 Apr 2019 14:34:
```

This allows the browser to then parse the actual data that's attached to the response as HTML code.

HTML is the core "programming language" (technically, it's not a programming language - you can't write any logic with it) of the web. HTML stands for "Hyper Text Markup Language" and it describes the structure of a webpage.

Code Example

The code looks like this:

```
<h1>Breaking News!</h1>

<p>Websites work because browser understand HTML!</p>
```

<h1> and <p> are called **"HTML tags"**.

Every HTML tag has some semantic meaning which the browser understands, because HTML is also standardized. Hence there is no guessing about what a <h1> tag means.

The browser knows how to parse HTML and now simply goes through the entire response data (also called "the response body") to render the website.

Step 4 - Page Is Displayed

The browser goes through the HTML data returned by the server and **builds a website** based on that.

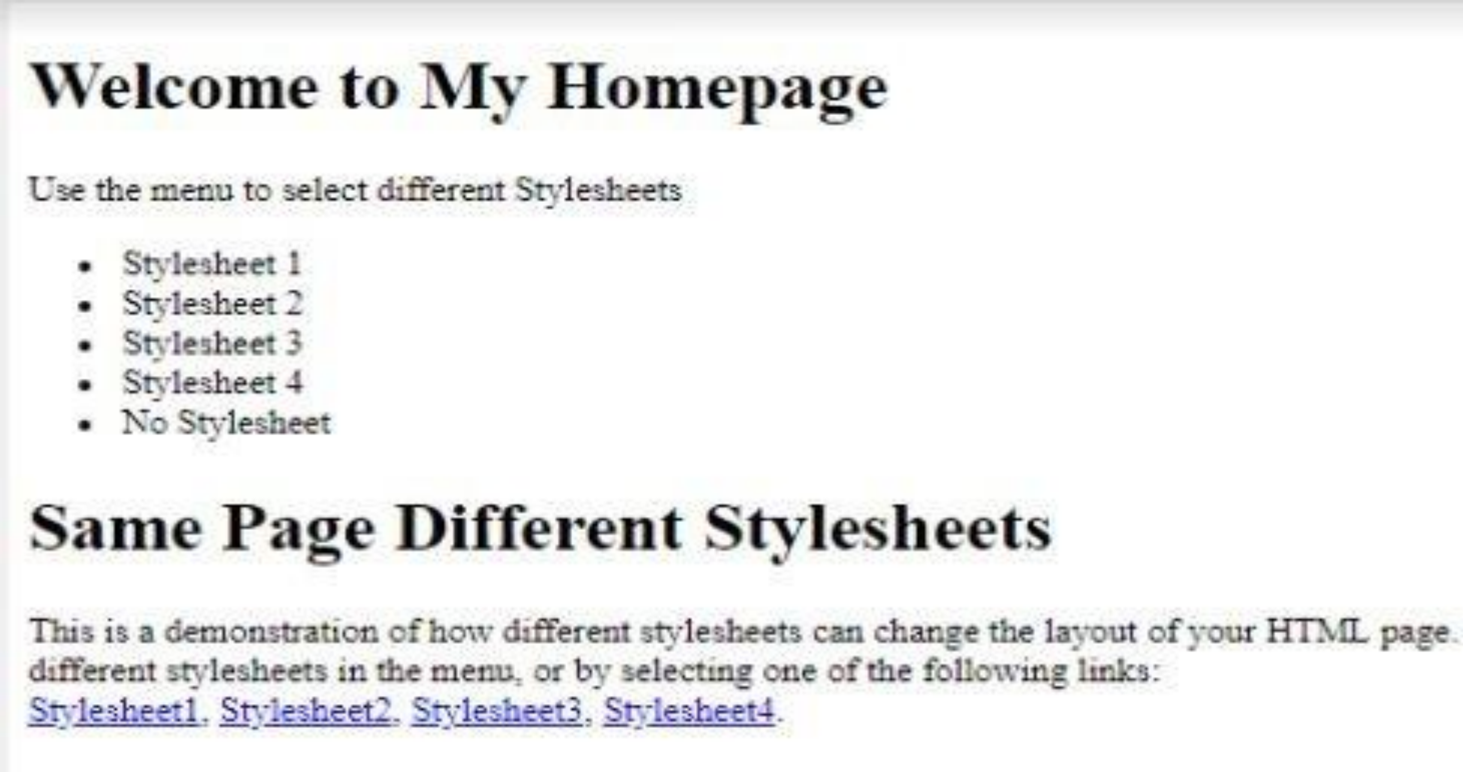
HTML does not include any **instructions** regarding what the site should **look like**

Example: how it should be styled.

It really only defines the structure and tells the browser which content is a heading, which content is an image, which content is a paragraph etc.

This is especially important for accessibility - screen readers get all the useful information out of the HTML structure.

A page that only includes HTML



Welcome to My Homepage

Use the menu to select different Stylesheets

- Stylesheet 1
- Stylesheet 2
- Stylesheet 3
- Stylesheet 4
- No Stylesheet

Same Page Different Stylesheets

This is a demonstration of how different stylesheets can change the layout of your HTML page.
different stylesheets in the menu, or by selecting one of the following links:
Stylesheet1, Stylesheet2, Stylesheet3, Stylesheet4.

CSS

Web page containing only HTML looks not that beautiful !

That's why there's another important technology **CSS** ("Cascading Style Sheets").

CSS is all about adding styling to the website. That is done via **"CSS rules"**:

Example:

```
h1 {
  color: blue;
}
```

This rule would color all <h1> tags blue.

CSS

CSS Rules can be added inside of the HTML code but typically, they're part of **separate .css files** which are requested separately.

A website can be made up of more than the data of the first response we get.

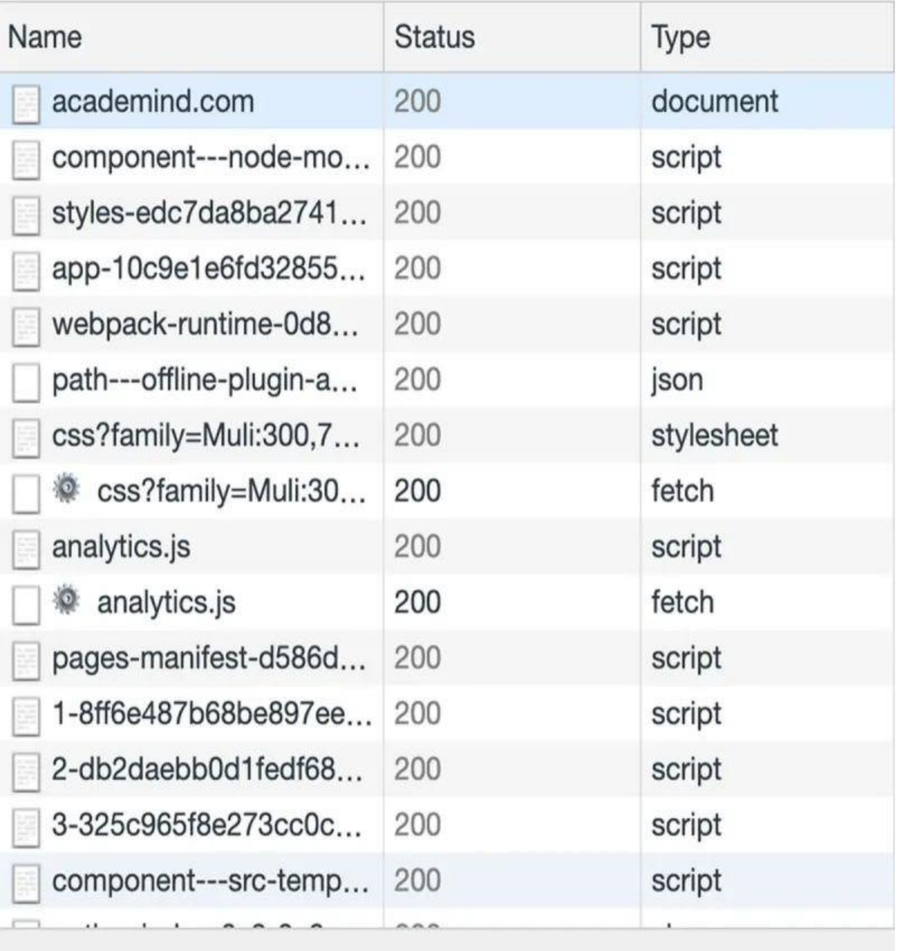
In practice, websites fetch a lot of additional data (via additional requests and responses) which are kicked off once the first response arrived.

How does that work?

HTML code of the first response simply contains instructions to fetch more data via new requests - and the browser understands these instructions.

Fetching of Additional Data

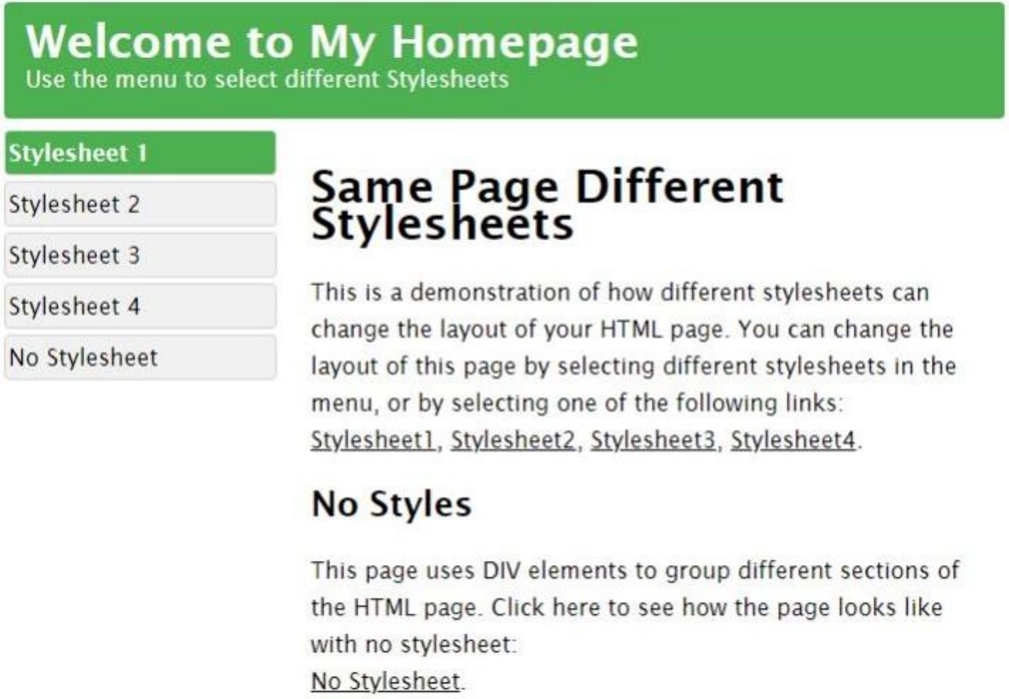
```
<link rel="stylesheet"
href="/page-styles.css" />
```



Name	Status	Type
academind.com	200	document
component---node-mo...	200	script
styles-edc7da8ba2741...	200	script
app-10c9e1e6fd32855...	200	script
webpack-runtime-0d8...	200	script
path---offline-plugin-a...	200	json
css?family=Muli:300,7...	200	stylesheet
⚙ css?family=Muli:30...	200	fetch
analytics.js	200	script
⚙ analytics.js	200	fetch
pages-manifest-d586d...	200	script
1-8ff6e487b68be897ee...	200	script
2-db2daebb0d1fedf68...	200	script
3-325c965f8e273cc0c...	200	script
component---src-temp...	200	script

CSS

Together with CSS, the browser is able to display webpages like this



Welcome to My Homepage

Use the menu to select different Stylesheets

Same Page Different Stylesheets

This is a demonstration of how different stylesheets can change the layout of your HTML page. You can change the layout of this page by selecting different stylesheets in the menu, or by selecting one of the following links: Stylesheet1, Stylesheet2, Stylesheet3, Stylesheet4.

No Styles

This page uses DIV elements to group different sections of the HTML page. Click here to see how the page looks like with no stylesheet:
No Stylesheet.

JavaScript

There is another programming language involved (really is a programming language!): JavaScript.

It's not always visible but all dynamic content you find on a website (e.g. tabs, overlays etc.) is actually only possible because of JavaScript.

It allows web developers to define **code that runs in the browser** (not on the server), hence JavaScript can be used to change the website while the user is viewing it.

Conclusion

These are the four steps that are always involved when you enter a page address like svnit.ac.in and you thereafter see the website content in your browser.

1. URL Gets Resolved
2. Request Is Sent
3. Response Is Parsed
4. Page Is Displayed

Server-side

Languages that don't work in the browser but that can run on a normal computer (a server is in the end just a normal computer).

Examples would be: **Node.js, PHP, Python**

Important: With the exception of PHP, you can also use these programming languages for other purposes than web development.

Node.js is indeed primarily used for server-side programming (though it's technically not limited to that), Python is also very popular for data science and machine learning.

Browser-side

In the browser, there are exactly three languages/ technologies that are used.

These three browser-side languages are all mandatory to know and understand:

HTML (for the structure)

CSS (for the styling)

JavaScript (for dynamic content)

People often use the words "the Internet" and "the web“ interchangeably. Is there a difference?

- The World Wide Web is the system we use to access the Internet.
- Email and instant messaging are other systems we use to access the Internet.
- We use programs called ***web browsers*** to use the World Wide Web for a wide variety of purposes.

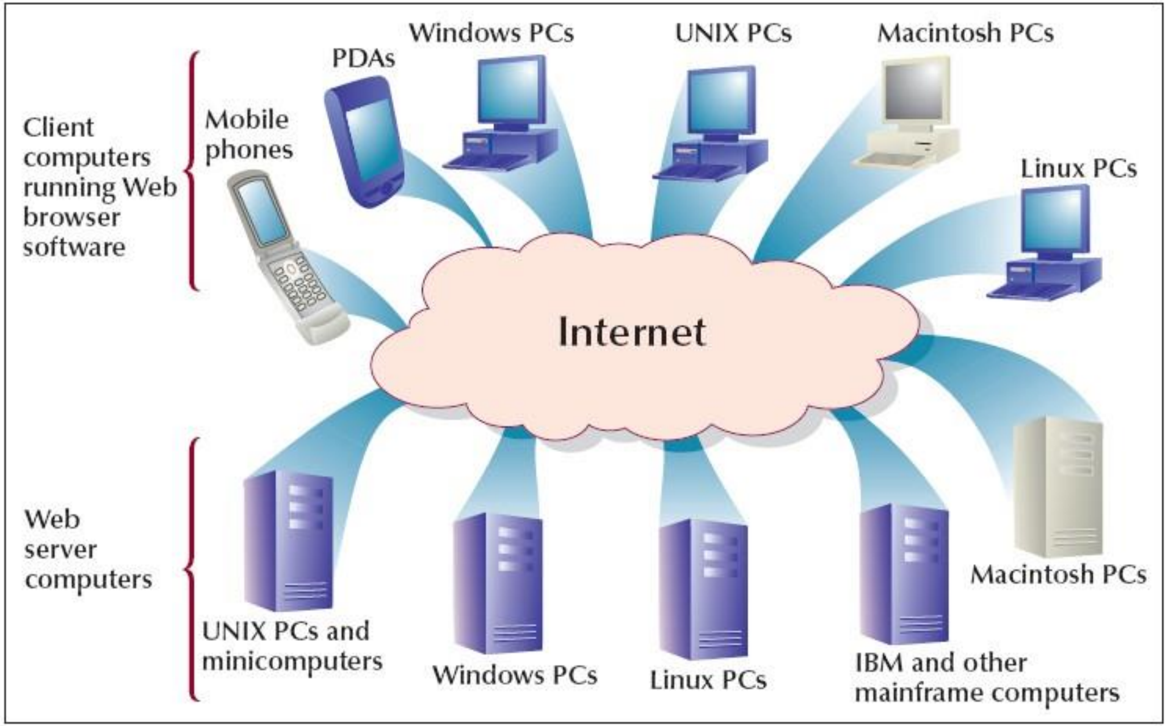
- When you use your Internet connection to become part of the Web, your computer becomes a Web client in a worldwide network based on a client/server network (architecture).

- Web browsers are software that you run on your computer to make it work as a Web client

- is a computer programs that delivers (serves) content web pages, using the Hypertext Transfer Protocol HTTP.

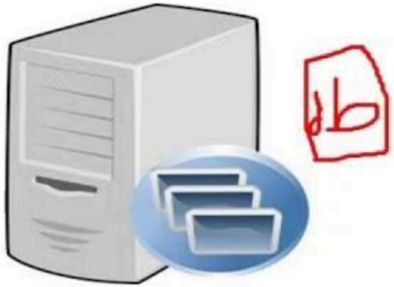
- The term web server can also refer to a computer connected to the Internet that contains files their owners have made available publicly through their Internet connections.

Further info : link
link2



Server types:-

Web server

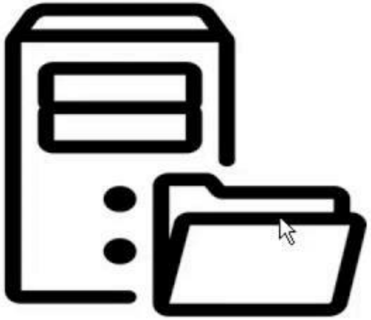


application server

Database server

mail server

DNS server



FTP server

Types of Web Servers

- Apache HTTP Server which occupies (69.01%) of the general use. It is developed in 1994 by Rob McCool.

- It is Available for free charge.

- It Runs on operating systems including FreeBSD-UNIX, HP-UX, Linux, Microsoft Windows, SCO- UNIX and Solaris

- Microsoft Internet Information Server (IIS) which occupies (23.26%) of the general use.

- It comes bundled with Microsoft Windows Server operating system.

- It runs on Windows 2022 , 2019,2016 etc. server OS.
- Sun Java System Web Server (JSWS) (former names are Sun One, iPlanet Enterprise Server and Netscape Enterprise Server) (0.86%)

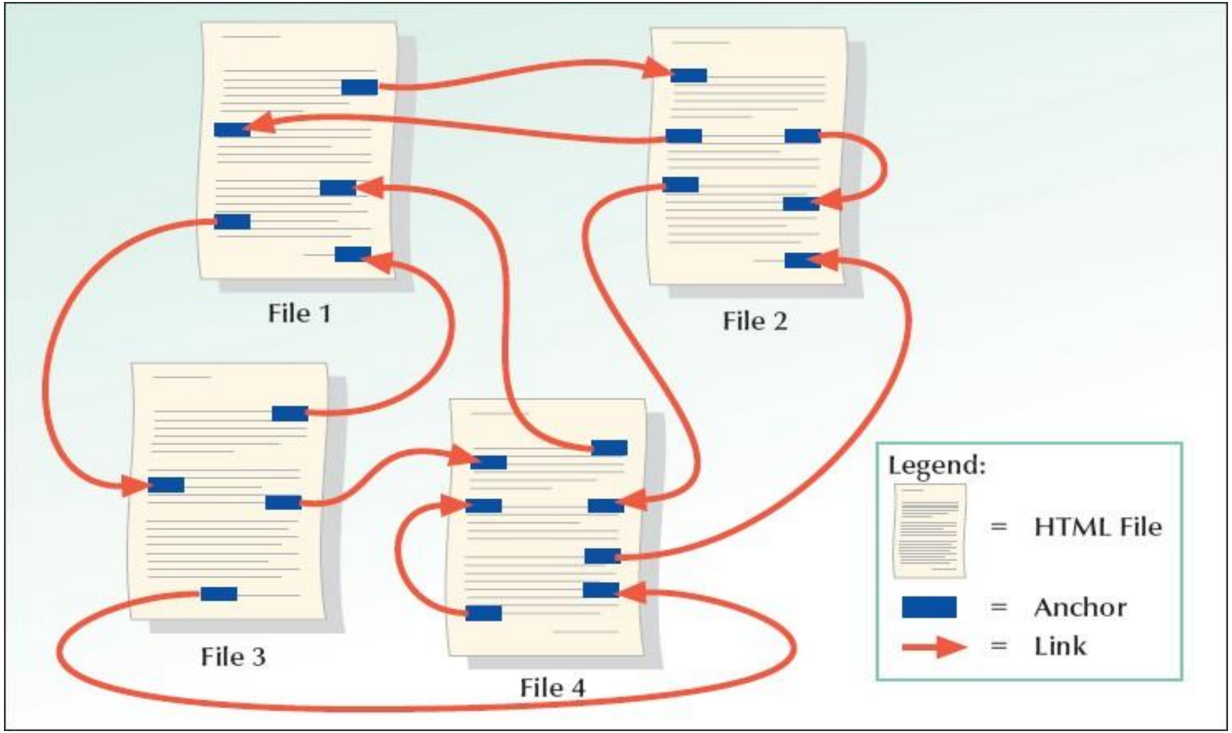
- HTML is a standard language

- HTML uses codes (tags) to tell the Web browser software how to display text.

- HTML document is defined as a text file that contains HTML tags.

- When a Web browser displays an HTML document, it is referred to as a Web page.

- One of the most important features of html is HTML anchor tag - to create hyperlinks in webpages.
  <a> </a>
- create Hypertext links.
- Hypertext links can connect HTML documents together or can connect one part of HTML document to another part.
- *When hyperlinks connect to Multimedia files , it is called Hypermedia links.*



HTML

- Having a collection of linked Web pages with a common theme or focus is called a website.

- Each website should have a main or home page to start the web site from.

- Each computer on the Web is given a unique identification number called Internet Protocol Address or (IP).

- Remembering IP addresses are hard to remember
- IP address may also be assigned a host or domain name.

- Domain name are unique name associated with specific IP address by a program that runs on an Internet host computer.

- This program is called **DNS (Domain Name System) software**.

- Domain Name Software or (DNS) is an Internet service that translates domain names into IP addresses.

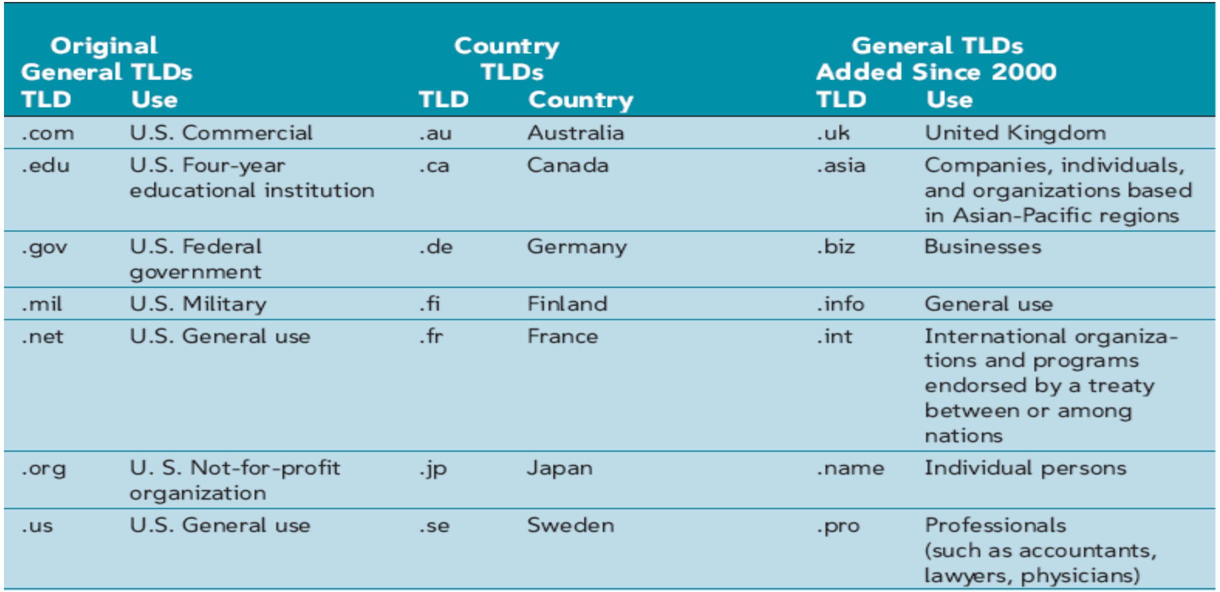
- Every time you use a domain name, a DNS service must translate the name into the corresponding IP address.

- For example, the domain name **www.example.com** might translate to **198.105.232.4**.

- The host computer that runs the DNS service or software is called Domain name server or **DNS server**.

Domain Name Addressing

- The last part of domain name is called its top-level domain (TLD).



Original General TLDs		Country TLDs		General TLDs Added Since 2000	
TLD	Use	TLD	Country	TLD	Use
.com	U.S. Commercial	.au	Australia	.uk	United Kingdom
.edu	U.S. Four-year educational institution	.ca	Canada	.asia	Companies, individuals, and organizations based in Asian-Pacific regions
.gov	U.S. Federal government	.de	Germany	.biz	Businesses
.mil	U.S. Military	.fi	Finland	.info	General use
.net	U.S. General use	.fr	France	.int	International organizations and programs endorsed by a treaty between or among nations
.org	U. S. Not-for-profit organization	.jp	Japan	.name	Individual persons
.us	U.S. General use	.se	Sweden	.pro	Professionals (such as accountants, lawyers, physicians)

- How to tell the browser the needed information to

  allocate the wanted web site to be displayed.
- The Addressing schema = uniform resource allocator (URL)

- URLs, or Uniform Resource Locators, are the schema by which documents or data are addressed in the World Wide Web.

Addressing Schema

- **URL** contains the following information:
  - **Transfer protocol** to use when transporting the file
  - **Domain name** of computer on which file resides
  - **Pathname** of folder or directory on computer on which file resides
  - **Name** of the file

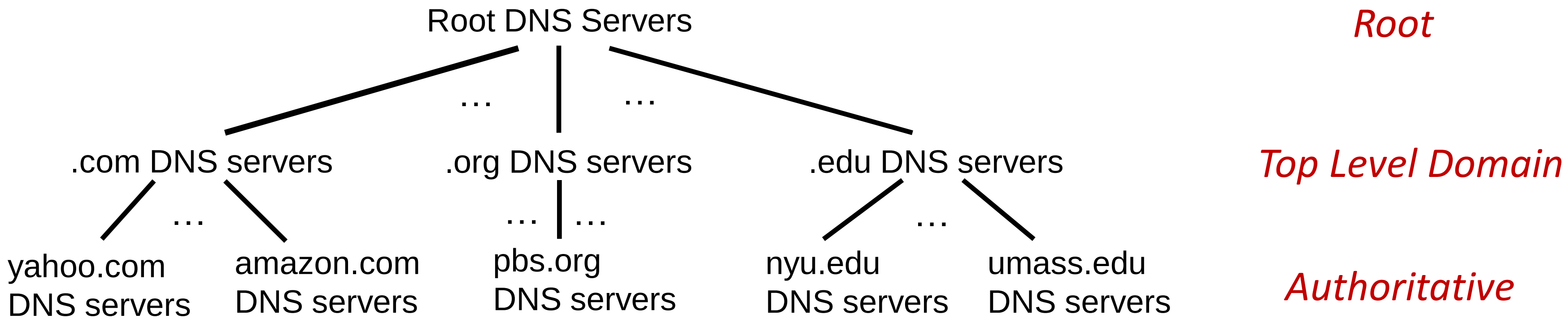
Domain Name System (DNS)

Domain Name System (DNS) is a hostname for **IP address** translation service. DNS is a distributed database implemented in a hierarchy of name servers. It is an application layer protocol for message exchange between clients and servers. It is required for the functioning of the Internet.

What is the Need of DNS?

Every host is identified by the IP address but remembering numbers is very difficult for people also the IP addresses are not static therefore a mapping is required to change the domain name to the IP address. So DNS is used to convert the domain name of the websites to their numerical IP address.

DNS: a distributed, hierarchical database

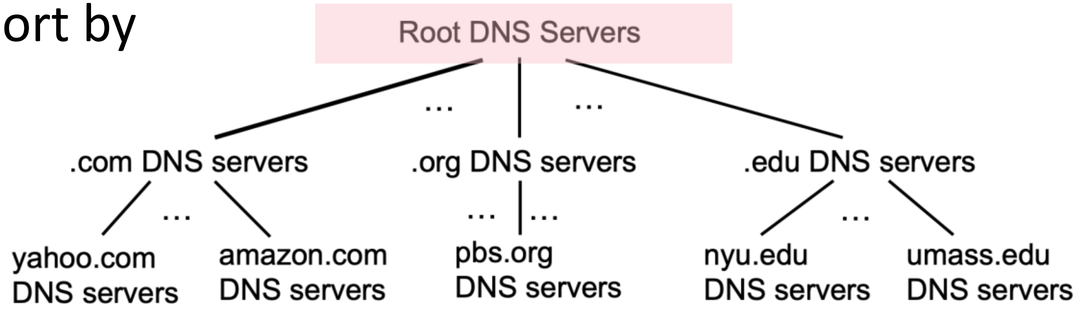


Client wants IP address for www.amazon.com; 1st approximation:

- client queries root server to find .com DNS server
- client queries .com DNS server to get amazon.com DNS server
- client queries amazon.com DNS server to get IP address for www.amazon.com

DNS: root Name Servers

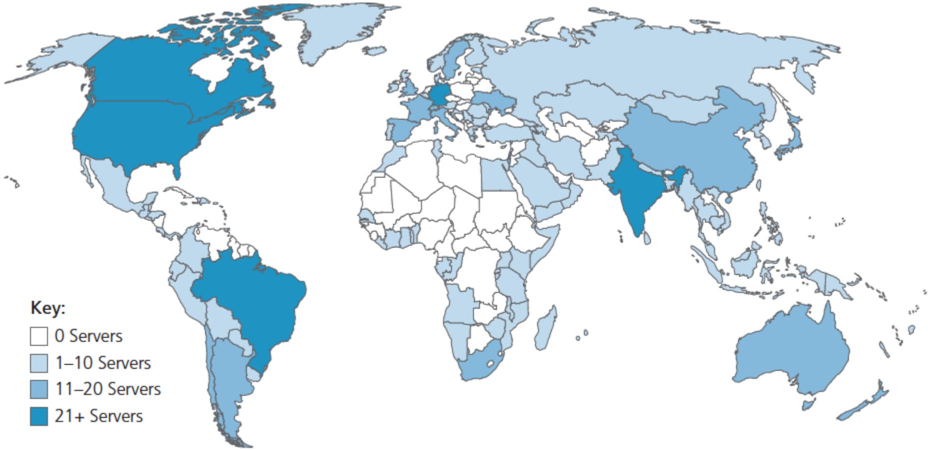
- official, contact-of-last-resort by name servers that can not resolve name



DNS: root Name Servers

- official, contact-of-last-resort by name servers that can not resolve name
- *incredibly important* Internet function
  - Internet couldn't function without it!
  - DNSSEC – provides security (authentication, message integrity)
- ICANN (Internet Corporation for Assigned Names and Numbers) manages root DNS domain

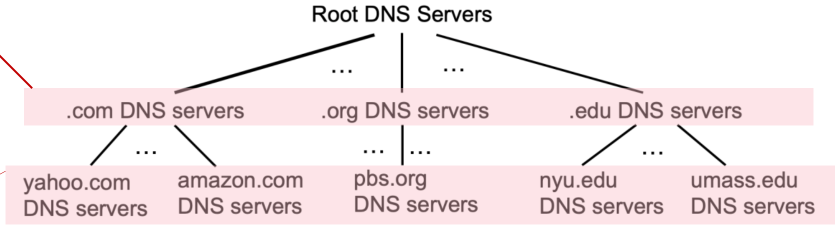
13 logical root name "servers" worldwide each "server" replicated many times (~200 servers in US)



Top-Level Domain, and authoritative servers

Top-Level Domain (TLD) servers:

- responsible for .com, .org, .net, .edu, .aero, .jobs, .museums, and all top-level country domains, e.g.: .cn, .uk, .fr, .ca, .jp
- Network Solutions: authoritative registry for .com, .net TLD
- Educause: .edu TLD

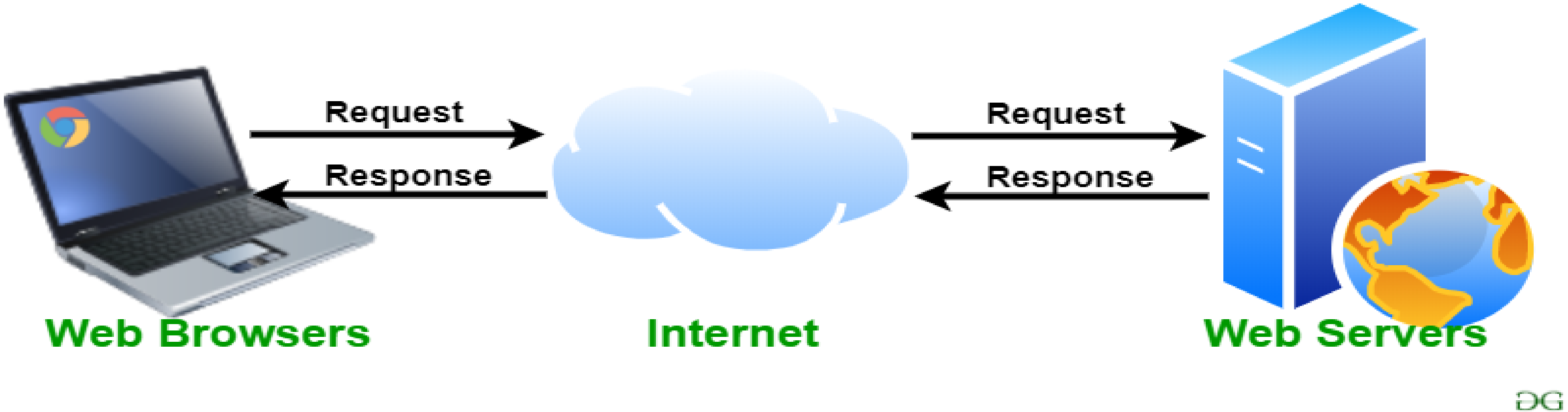


authoritative DNS servers:

- organization’s own DNS server(s), providing authoritative hostname to IP mappings for organization’s named hosts
- can be maintained by organization or service provider

- **Web Server:** Web server is a program which processes the network requests of the users and serves them with files that create web pages. This exchange takes place using Hypertext Transfer Protocol (HTTP).

- **Type of Web Server**
  1. Apache HTTP server
  2. Microsoft Internet Information Services (IIS)
  3. Lighttpd
  4. Jigsaw Server
  5. Sun Java System



Web Server:

Web servers are generally user space applications that run on top of an operating system and are generally not part of it. In general, an operating system is responsible for managing hardware resources and allocating them to applications.

In order to set up a Web server, you need a dedicated computer (PC or Macintosh) running Windows/95, Windows/NT, or Linux or a Macintosh computer running MacOS. You also need a direct Internet connection and TCP/IP software. You can download shareware HTTP software for these platforms and operate your own Web server.

- **Web Browser Definition:** A software application used to access information on the World Wide Web is called a Web Browser. When a user requests some information, the web browser fetches the data from a web server and then displays the webpage on the user's screen.
- **Web Documents :** Web Documents are often defined as a simple HTML files. A web document, such as this page, is often made up of several files and is accessed via a URL (uniform resource locator). The HTML file itself contains most of the times style sheets, background images, other graphics, Java programs, and other files.
- **Markup Language:** Markup is what HTML tags do to the text inside of them; they mark it as a specific type of text. For example, markup text could come in the form of boldface or italicized type to draw specific attention to a word or phrase.
- **HTML:** HTMLstands for Hyper Text Markup Language. HTML is the standard markup language for creating Web pages. HTML describes the structure of a Web page.

- **Hypermedia:** Hypermedia, an extension of the term hypertext, is a nonlinear medium of information that includes graphics, audio, video, plain text and hyperlinks. This designation contrasts with the broader term multimedia, which may include non-interactive linear presentations as well as hypermedia. Hypertext is text which contains links to other texts.
- **Web Site organization:** Site organization is more design focused and page specific. It refers to how the information on each page is placed and how that affects the user experience. At its simplest, a website provides a quick and easy way of communicating information between buyers and sellers.
- **Web Application:** A web application is software that runs in your web browser. Businesses have to exchange information and deliver services remotely. They use web applications to connect with customers conveniently and securely.
- **Web Interface:** A Web user interface or Web app allows the user to interact with content or software running on a remote server through a Web browser. The content or Web page is downloaded from the Web server and the user can interact with this content in a Web browser, which acts as a client.

- **What is Web Accessibility:** Web accessibility means that websites, tools, and technologies are designed and developed so that people with disabilities can use them. More specifically, people can: perceive, understand, navigate, and interact with the Web. contribute to the Web.

Thank you !