# Web Technologies

## Hyper Text Markup Language

HTML is used to describe the structure of most modern websites; it was first developed by Tim Berners-Lee in 1990 at Cern. It was first implemented in WorldWideWeb, the first web browser [1]. It is based on the Standard Generalized Markup Language, a tag-based language for describing the layout and structure of documents.

As HTML is a tag-based language, content is surrounded by tags which tell the interpreter how to render the content. I.e., Surrounding text with a <b> **tag will make the text bold.** </b>.

HTML has evolved over the years and is now maintained by The Web Hypertext Application Technology Working Group, which also now hosts the Living Standard for HTML (Widely referred to as HTML5) [2].

The WHATWG is a community of developers of all major web browsers, including Google, developers of Chrome, Apple, the developers of Safari, and Mozilla, the developers of Firefox. This group of companies help steer the continued development of HTML and other core web standards which helps ensure that Web Pages render in a consistent manner across a range of browsers and hardware.

### The Document Object Model (DOM)

The DOM is an application programming interface (API) which defines the structure of documents and data of a webpage, it provides an interface for JavaScript to dynamically modify the content of a webpage.

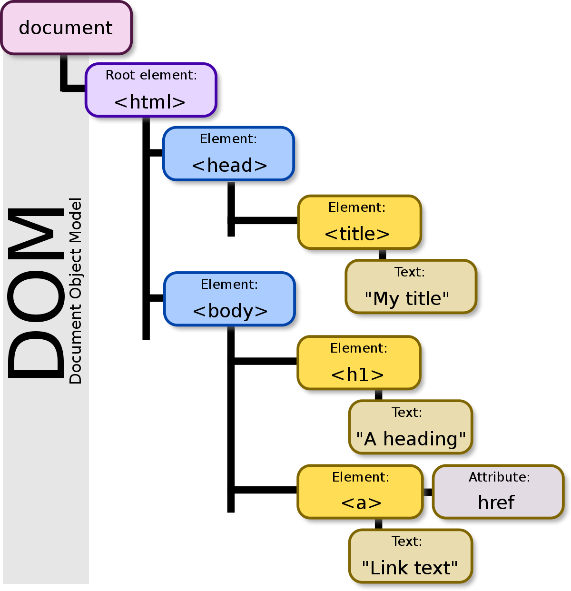
The DOM is created when a webpage is first loaded and is handled by the browser. It defines all the HTML elements, as well as the properties, methods and events associated with each of these objects.

Figure 1- Diagram of the DOM (Taken from https://en.wikipedia.org/wiki/File:DOM-model.svg)

### The Canvas

The Canvas is HTML5’s answer to how to render dynamic graphics on a webpage. And is how I plan on rendering graphics to the viewer in my simulation.

Unlike other implementations the Canvas does not require any additional technologies than those defined in the W3C standards, this allows for graphics to be rendered on any modern browser without any additional technologies, such as Microsoft’s Silverlight or Adobe’s Flash player; both of these technologies are now obsolete- partially because of the Canvas’ inclusion in the standard, allowing Canvas elements to be consistently rendered across all modern browsers, and even aboard spacecraft4].

#### Canvas optimisation via WebGL

WebGL is a JavaScript API for rendering high-performance interactive 2D and 3D graphics on the web, its technology is part of HTML5 and is used in all modern Web Browsers.[5]

As the Canvas element uses WebGL to render content to the user, rather than rendering the Canvas content itself, via the same processor threads allocated to render the rest of the webpage, it allows for hardware-based acceleration of graphics performance. This has been adopted in Chrome, Firefox, and Safari via WebKit [6].

WebGL allows browsers to harness the power of hardware designed for fast graphics performance, such as dedicated graphics cards which use the DirectX or Vulcan APIs, or the integrated graphics cores inside Apple’s M series of processors, which render using their Metal API. These graphics APIs are used for efficiently rendering complex 3D scenes commonly found in modern video games, and movies. And thanks to WebGL can also be used to render my sine waves slightly faster!

## Cascading Style Sheets (CSS)

Whilst HTML provides the structure of the document CSS provides the presentation, layout, and styling.

Rules are used to describe the desired styling of sets of elements in the DOM as they can be cascaded after each other with rules applied later taking precidence over rules defined earlier.

A rule such as

code fragment 1 - An example of cascading CSS.

p {  
 colour: red;  
}

.description {  
 colour: black;  
}

Would set the text colour of each paragraph element to be red throughout the entire DOM, except any element with the class ‘description’ which would have its text colour set to black.

### Flexbox

The modern Flexbox layout which was introduced to CSS in 2017[7] aims to provide an efficient way to lay out objects so that they are best able to be shown to users regardless of their screen resolution.

![A picture containing text

Description automatically generatedDynamically scaling webpage content is increasingly important due to the rise of smartphones and people increasing viewing content on mobile devices, often with a vastly different display ratio than the standard 16:9 of computer monitors. Flexbox helps to aid in this by allowing content to resize and reorganise itself to be able to fit inside the display container, without overflowing. Flexboxes allow for elements to wrap around and be displayed in a single-column view on vertical displays whilst displaying as a row on a horizontally oriented display.

Figure 2 – Flexbox example, the same content will be dynamically adjusted to best fit depending on display ratio

This is a technique that I plan on employing in my project to ensure that the visualisation will be able to display correctly regardless of the screen resolution the user is viewing the content on.

## JavaScript

JavaScript is a general-purpose scripting language that forms the final of the three core web technologies, whilst HTML and CSS define the layout and content of a webpage JavaScript adds functionality and dynamic behaviour to the page.

It is implemented using the ECMAScript standard on all modern web browsers which is why webpages can function, mostly similarly, regardless of the browser used.

It was first created in 1995 by Brendan Eich [8], whilst he was working on the Netscape Navigator browser. Eich went on to co-found the Mozilla project, which is responsible for development of the Firefox web browser.

### What is the difference between Java and JavaScript?

Everything.

JavaScript is only called such due to software engineers generally being terrible at branding. JavaScript was originally going to be called Mocha, then was known as LiveScript (A much better name), and finally settled on being referred to as JavaScript due to it being able to perform similar tasks on the web as Java Applets, see below.

Unlike Java, JavaScript is a weakly typed language which means variables do not need to be given an explicit type and the same variable could be used to store the value of PI or the combined works of Shakespeare, and the browser will have little idea of what will happen when it is told to square the variable and will be very confused when it doesn’t know what to do when it is told to multiply Hamlet by Macbeth.

To solve this problem Microsoft developers created TypeScript in 2012 [9], it contains syntax for types and TypeScript-supported code editors will raise a warning when it sees issues with how variables are intended to be used, however, it will eventually be turned back into JavaScript before being executed so only exists for the sanity of software developers.

JavaScript is an interpreted scripting language[[1]](#footnote-1). Java on the other hand is a compiled language, but rather than being complied straight to machine code it is compiled to bytecode which can be executed by a Java Virtual Machine (JVM), this has the advantage that only one executable is needed regardless of the processer architecture of the device you intend on executing the code on but does mean the processer needs a JVM in order to run the software.

#### Java Applets

Graphical user interface

Description automatically generated with low confidenceJava was used on the web around the time when JavaScript was being developed to dynamically display content via Java Applets [10], which were small applications that complied to Java byte code before being displayed to the user on a webpage, they ran much faster than JavaScript due to being complied rather than interpreted and because of this they were used for computationally intensive tasks, such as calculating and visualising the Mandelbrot set

Java applets were not widely successful [12] but may have been many years ahead of their time, as WebAssembly uses a very similar principle to Java Applets to perform computationally intensive tasks by compiling software into byte code before being executed.

Figure 3 – A Java Applet displaying a section of the Mandelbrot set (taken from https://commons.wikimedia.org/wiki/File:Mandelbrot\_java\_applet.png)

### ECMAScript

ECMAScript, with ECMA standing for the European Computer Manufacturers Association, defines the standard for how the JavaScript programming language should behave [13]. Different browsers implement different features of the specification, but all modern browsers have implemented ES6 (ECMAScript 6), although since 2015 the standard has been renamed to ECMAScript 2015, with subsequent versions incrementing the year.

Notably, Internet Explorer has not implemented most features of ES6 which is why most modern web applications fail to run on Internet Explorer.

## WebAssembly

Finally nearing the present day, WebAssembly (Wasm) is a technology adopted in 2019 by the World Wide Web Consortium, it is notable as the second standard for executing code in all modern browsers.[14]

WebAssembly allows for code to be written in many languages and then compiled into a low-level binary format. This would theoretically allow for massive computational time savings when delivering computationally intensive programs to the user. This is because you can write the code in an extremely high-performance language, like C, and convert it to a binary format for the user to run on their machine far faster than JavaScript could ever run.  
  
As JavaScript, was not designed to be a high-performance language and is compiled by a Just in Time compiler ahead of execution it would not be the language of choice for something computationally intensive, such as a Fast Fourier Transform. WebAssembly on the other hand would be complied to binary code far ahead of time and would be downloaded to the user’s machine when they load the web page and be immediately ready for fast execution.  
  
I am planning to use the Fourier Transform in my demo to convert a function from the time to frequency domain. This is a very computationally intensive operation, given the large number of sample points needed to reach the Nyquist frequency of the data that a user could generate and therefore could utilise the performance advantages of WebAssembly to have the function be programmed in an appropriate language, such as C, and then run by the user's browser, showing the user the output theoretically much quicker than with JavaScript.

A popular design tool called Figma [15] is currently doing this to deliver an extremely computationally intensive design tool to users via a web browser. Previously something this powerful would only be able to run via a native, compiled desktop application, such as Photoshop.

[1] T. Berners-Lee The WorldWideWeb browser W3.org [Online]. Available: <https://www.w3.org/People/Berners-Lee/WorldWideWeb.html>   
[Accessed: 11 November 2022].

[2] Web Hypertext Application Technology Working Group *HTML Living Standard* [Online]. Available: <https://html.spec.whatwg.org/> [ Accessed: 11 November 2022].

[3] Jonathan Robie, Texcel Research *What is the Document Object Model* W3.org [Online]. Available: <https://www.w3.org/TR/WD-DOM/introduction.html> [Accessed: 24 November 2022].

[4] Lithios *A Look Under the Hood of SpaceX’s Dragon Capsule* [Online].  
Available: <https://lithiosapps.com/a-look-under-the-hood-of-spacexs-dragon-capsule/> [Accessed 11 November 2022]

[5] Mozilla *WebGL: 2D and 3D graphics for the web* [Online]. Available: <https://developer.mozilla.org/en-US/docs/Web/API/WebGL_API>   
[Accessed: 11 November 2022].

[6] Jen Simmons *New WebKit Features in Safari 15* [Online]. Available: <https://webkit.org/blog/11989/new-webkit-features-in-safari-15/> [Accessed: 11 November 2022].

[7] W3C Working Group *CSS Snapshot 2017* [Online]. Available: <https://www.w3.org/TR/css-2017/> [Accessed: 24 November 2022].

[8] T. J. DeGroat *The History of JavaScript: Everything You Need to Know* [Online]. Available: <https://www.springboard.com/blog/data-science/history-of-javascript/> [Accessed: 24 November 2022].

[9] Jeff Martin *TypeScript Released to Improve JavaScript Development* [Online]. Available: <https://www.infoq.com/news/2012/10/typescript/> [Accessed: 24 November 2022].

[10] Mozilla *The Embedded Java Applet element* [Online]. Available: <https://developer.mozilla.org/en-US/docs/Web/HTML/Element/applet> [Accessed: 24 November 2022].

[11] Lin Clark *A crash course in just- in- time (JIT) compilers* [Online]. Available: <https://hacks.mozilla.org/2017/02/a-crash-course-in-just-in-time-jit-compilers/> [Accessed: 24 November 2022].

[12] Michael Byrne *The Rise and Fall of the Java Applet: Creative Coding’s Awkward Little Square* [Online]. Available: <https://www.vice.com/en/article/8q8n3k/a-brief-history-of-the-java-applet> [Accessed: 24 November 2022].

[13] ECMA International *ECMAScript® 2023 Language Specification* [Online]. Available:  
<https://tc39.es/ecma262/#sec-intro> [Accessed: 24 November 2022].

[14] Mozilla *WebAssembly* [Online]. Available: <https://developer.mozilla.org/en-US/docs/WebAssembly> [Accessed: 24 November 2022].

[15] Evan Wallace CTO Figma *WebAssembly cut Figma’s load time by 3x* [Online]. Available: <https://www.figma.com/blog/webassembly-cut-figmas-load-time-by-3x/> [Accessed: 24 November 2022].

1. Theoretically at least, in reality, most JavaScript ran on a modern web browser is compiled ahead of time using a Just in Time compiler [11] meaning it is compiled as close to run- time as possible. [↑](#footnote-ref-1)